

Piia-Maria Sinkkonen

INDICO -VERKKOPORTAALIN KÄYTTÖLIITTYMÄN MUUTOSTYÖT

Opinnäytetyö
Tietotekniikan koulutusohjelma


Huhtikuu 2010




MIKKELIN AMMATTIKORKEAKOULU

Mikkeli University of Applied Sciences

KUVAILULEHTI

 <p>MIKKELIN AMMATTIKORKEAKOULU Mikkeli University of Applied Sciences</p>		Opinnäytetyön päivämäärä 30.4.2010	
Tekijä(t) Piia-Maria Sinkkonen		Koulutusohjelma ja suuntautuminen Tietotekniikan koulutusohjelma Ohjelmisto- ja mediatekniikan sv.	
Nimeke Indico- verkkoportaalin käyttöliittymän muutostyöt			
Tiivistelmä <p>Tämä opinnäytetyö käsittelee Indico- verkkoportaalin käyttöliittymään suoritettua parannustyötä. Insinöörityön käytännön osuus on toteutettu Euroopan hiukkastutkimuskeskus CERNin alaisuudessa loppukesästä 2008.</p> <p>Nykyään monilla instituutioilla ja tapahtumilla on oma ennalta määrätty visuaalinen ulkoasu. Verkkosivuja tarjoavilta palveluilta tämä vaatii sivujen ulkoasun ja mahdollisesti myös rakenteen muokattavuutta. Muotoilumahdollisuuksien vähyys voi johtaa siihen, että tapahtumajärjestäjät turvautuvat omiin, palvelun ulkopuolisiin verkkosivuihin ulkoasun kunnollista esittämistä varten.</p> <p>Tehty työ on osa laajempaa Indico-projektia. Indico on Python –skriptikielellä toteutettu selainpohjainen sovellus tapahtumien ja tilojen hallinnointiin. Osaprojektin tavoitteena oli suunnitella ja toteuttaa Indico-verkkoportaalin tapahtumaosion (conference pages) käyttöliittymän muutos- ja parannustyöt. Työn erityispainona oli tapahtumasivujen käyttäjälähtöisten muutosten mahdollistaminen ja siten saada järjestelmä vastaamaan paremmin tapahtumanjärjestäjien tarpeita.</p> <p>Tehtävänannon ratkaisuna suoritettu työ vaati vanhan HTML-rakenteen uudistamista ja erillisten CSS-tiedostojen määrittelyn ja toteutuksen. HTML- ja CSS- tiedostoja varten luotiin niitä tukevat toiminnallisuudet Python- kieltä käyttäen. Python- kielen perusteet on rajattu tämän opinnäytetyön ulkopuolelle, koska työ painottuu käyttäjälähtöisten muutosten mahdollistamiseen, ja tehdyt ominaisuudet on toteutettavissa muitakin skriptikieliä käyttäen. Projektin edetessä syntyivät myös kuvien lataamisen ja käyttöönnoton mahdollistavat ominaisuudet. Projektin työnannon lisäksi toteutettiin erillinen esikatselu- ominaisuus sivun laatijoiden työn helpottamiseksi.</p> <p>Työn teoriaosuudessa alussa perehdytään verkkosovellusten perusteisiin sekä HTML- ja CSS- kieliin. Tämän jälkeen tarkastellaan, kuinka HTML- ja CSS- tekniikoita hyödyntämällä voidaan vaikuttaa verkkosivun rakenteeseen ja ulkoasuun. Teoriaosuuden lopussa esitellään erilaisia työvälineitä, joita käytetään verkkosovelluksia toteutettaessa. Lisäksi kerrotaan eri selainten huomioonottamisen tärkeydestä. Kirjallisen osuuden loppu on varattu käytännön työn selostamiseen ja johtopäätöksille.</p>			
Asiasanat (avainsanat) HTML, CSS, taulukoton taitto, cross-browser, Indico, CERN, WWW, verkkosivu			
Sivumäärä 54 s. + liitteet 3 s.	Kieli Suomi	URN http://www.urn.fi/URN:NBN:fi:amk-201005108573	
Huomautus (huomautukset liitteistä)			
Ohjaavan opettajan nimi Reijo Vuohelainen		Opinnäytetyön toimeksiantaja Euroopan hiukkastutkimuskeskus - CERN	

DESCRIPTION

 <p>MIKKELIN AMMATTIKORKEAKOULU Mikkeli University of Applied Sciences</p>		Date of the bachelor's thesis 30 th April 2010			
Author(s) Piia-Maria Sinkkonen		Degree programme and option Information Technology			
Name of the bachelor's thesis Customization of Indico's conference web portal					
Abstract <p>This bachelor's thesis was about the improvements made to the user interface of Indico web portal. The practical part of the thesis was carried out as an assignment by CERN, the European Organization for Nuclear Research in the late summer of 2008.</p> <p>Today, many institutions and events have their own predetermined visual appearance. This requires a customizable appearance of the web pages, creating its challenges for the web pages that provide services for those events. The lack of design possibilities could lead to a situation where the event organizers rely on their own web pages for the presentation.</p> <p>The work was a part of a broader Indico project. Indico is a browser-based web application for event management, implemented with the Python script language. The subproject's main task was to design and implement improvements to the interface of Indico's conference pages. The main focus was to make possible user-driven changes to the conference pages, and thus meet the needs of the organizers of the events.</p> <p>The theoretical section focused on the basics of web applications, and HTML and CSS techniques. It also described HTML and CSS techniques used in the webpage layout creation. The end of the theoretical part presented various tools used in implementation of web applications. One chapter also indicated the importance of multi-browser compatibility. The end of the thesis presented the practical work and the conclusions.</p> <p>The solution to the assignment required refactoring the old HTML structure and creating separate CSS files. Functions for the HTML and CSS files were created by using the Python language. The basics of Python were excluded, since the work focused on permitting the user-driven changes. Also, it was possible to create all the functions by using other scripting languages. During the project, a need appeared to create a possibility to upload images and deploy them. To do this, specific functions for picture uploading were created. In addition to the assignment, a separate page preview feature was created to help the authors of the event pages.</p>					
Subject headings, (keywords) HTML, CSS, tableless design, cross-browser, Indico, CERN, web page, WWW					
Pages 54 p. + appendices 3 p.		<table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td style="width: 50%; vertical-align: top;"> Language Finnish </td> <td style="width: 50%; vertical-align: top;"> URN http://www.urn.fi/URN:NBN:fi:amk-201005108573 </td> </tr> </table>		Language Finnish	URN http://www.urn.fi/URN:NBN:fi:amk-201005108573
Language Finnish	URN http://www.urn.fi/URN:NBN:fi:amk-201005108573				
Remarks, notes on appendices					
Tutor Reijo Vuohelainen		Bachelor's thesis assigned by CERN – the European Organization for Nuclear Research			

SISÄLTÖ

1	JOHDANTO	1
1.1	CERN.....	3
1.2	Indico	4
2	TEORIA	6
2.1	Verkkosovelluksen perusteet	6
2.1.1	Verkko ympäristönä	6
2.1.2	Kuinka asiakas-palvelin-malli toimii	6
2.1.3	Lyhyesti tilakoodista	8
2.1.4	URL:n rakenne	8
2.2	HTML	9
2.2.1	HTML kehityshistoria	9
2.2.2	Merkintä	11
2.2.3	Staattinen verkkodokumentti	14
2.2.4	Dynaamiset verkkosivut	14
2.3	CSS-tyylisivut	14
2.3.1	CSS:n kehityshistoria	16
2.3.2	CSS tyylien käyttö	17
2.3.3	CSS –merkintä	17
2.3.4	CSS -valitsimet	18
2.3.5	Kaskadisuus ja periytyvyys	19
2.3.6	Erillinen (ulkoinen) tyylitiedosto	20
2.4	Verkkosivun rakenteen luominen	20
2.4.1	Asemointi	21
2.4.2	Taulukkotaitto	23
2.4.3	Taulukoton taitto (tableless design)	26
2.4.4	Kelluva rakenne	30
2.4.5	Taulutaitto vs. taulukoton taitto	30
2.4.6	Eri selainten huomioon ottaminen	32
2.4.7	Työkalut sivuston teon apuna	33

3	CASE STUDY – CUSTOMIZATION OF INDICO’S CONFERENCE WEB	
	PORTAL	35
3.1	Indico versio 96.2	35
3.1.1	Konferenssi osion sivuston vanha rakenne	35
3.1.2	Konferenssi osion sivuston vanha asemointi	36
3.1.3	Konferenssi osion sivuston vanha muokkaus	37
3.1.4	Ongelmat vanhassa versiossa.....	38
3.2	Ratkaisuvaihtoehdot	38
3.2.1	Taustatyö ja tutkimus.....	39
3.2.2	Erilaisia ratkaisumalleja.....	39
3.2.3	Toteutettu ratkaisu	40
3.3	Tehdyt muokkaukset uuteen versioon	40
3.3.1	Tehdyt muokkaukset konferenssisivujen HTML-rakenteeseen.....	40
3.3.2	Tehdyt muokkaukset hallinnointisivuille.....	41
3.3.3	Oletusarvoisen rakenteen ja ulkoasun CSS-tiedosto.....	43
3.3.4	Esivalmistetut rakennepohjat	44
3.3.5	Funktiot CSS:n käyttöönottoa varten.....	46
3.3.6	Esikatselusivu (Preview page)	46
3.3.7	Esikatselusivua varten luodut funktiot.....	47
3.3.8	Kuvien latausta varten luodut toiminnallisuudet	47
3.4	Testaus	48
3.5	Turvallisuus	49
4	JOHTOPÄÄTÖKSET	50
4.1	Projektin lopputulokset.....	50
4.2	Indico- projektin tulevaisuus	50
4.3	Indico versio 0.97b2	51
4.4	Loppusanat.....	52
	LÄHTEET	53
	LIITTEET	

Kiitokset

Haluan kiittää seuraavia henkilöitä:

Jose Benito Gonzalez Lopez, Indico projektijohtaja ja ohjaajani CERNissä,

Thomas Baron:ia, osaston johtaja,

Jose Pedro Ferreira, kollega Indico työryhmässä

David Martin Clavo, kollega Indico työryhmässä.

Haluan kiittää heitä tästä ainutlaatuisesta mahdollisuudesta työskennellä Indico-projektin parissa. Lisäksi haluan kiittää kaikesta avusta, neuvoista ja tiedoista, joita sain työprojektini aikana. Oli upea kokemus työskennellä nämä kuukaudet osana CERNiä ja UDS-AVC osastoa.

Haluan myös kiittää CERNin kesäopiskelijatyöryhmää. Heidän panostuksensa teki minun CERNissä oleskelun mahdolliseksi.

Lisäksi haluan kiittää koulutusvastaavaa Reijo Vuohelaista, joka toimi tämän insinöörityön ohjaajana ja valvojana kouluni Mikkelin Ammattikorkeakoulun puolesta.

Lopuksi haluan kiittää Mikkelin Ammattikorkeakoulua taloudellisesta tuesta, joka teki osaltaan työskentelyn CERNissä mahdolliseksi.

Tässä työssä esiintyvää termistöä:

AJAX	(Asynchronous JavaScript and XML). AJAX ei ole ohjelmointikieli, vaan tapa käyttää jo olevia standardeja. AJAX mahdollistaa tiedonsiirron palvelimelta ja sivun osien päivityksen ilman, että koko sivu täytyy päivittää.
CERN	<i>Conseil Européen pour la Recherche Nucléaire</i> Sveitsin ja Ranskan rajalla sijaitseva Euroopan hiukkastutkimuskeskus.*
CSS	Kaskadisett tyylisheetit (Cascading Style Sheet) on tapa HTML-sivujen ulkoasun määrittelyyn koskematta itse HTML-tiedostoon.
CSS-tiedosto	Erillinen tiedosto, joka sisältää tyylisheetit sivun ulkoasun määrittelyyn.
Drag and drop	Raahaa ja pudota. Rakenteellisten osien siirtämistä hiiren avulla raahaten
HTML	HTML (HyperText Markup Language) on yksinkertainen sivunkuvaus- ja ohjelmointikieli, jolla luodaan eri ympäristöihin sopivia hypertekstiasiakirjoja ja se on kehitetty 1990-luvun alussa.**
HTML-tiedosto	ASCII-tiedosto, johon upotetut koodit tarkoittavat muotoiluja ja hypertekstilinkkejä.**
Indico	Indico (Integrated digital conferencing) on kokousten ja tapahtumien järjestämiseen tarkoitettu ohjelma. Indico on osa EU:n rahoittamaa projektia.

Tagi	Alku- tai lopputunniste eli tagi (tag) on HTML-elementin osa, joka kirjoitetaan karkisulkeismerkkien (< >) sisään.
URL	Uniform Resource Locator, standardi, jolla määritellään resurssin sijainti verkossa, esimerkiksi Internetissä.**
WWW	World Wide Web. CERN-tutkimuskeskuksessa englantilaisen Tim Berners-Leen vuonna 1989 kehittämä palvelu, joka tähtää maailmanlaajuiseen multimedia ja hypertekstitiedon välittämiseen HTTP-protokollan avulla Internet-verkossa.**
W3C	WWW Consortium. Ylläpitää ja hallinnoi WWW, CSS, XML ja HTML- standardeja.
WYSIWYG	What you see is what you get. Minkä näet sen saat. DTP-tekniikan termi, joka tarkoittaa, että kuvaruudulla näkyvä tulostettava sivu layout-mielessä voidaan tulostaa sellaisena kuin se ruudulla näkyy.**
XML	XML (eXtensible Markup Language), on merkintäkieli, joka on tarkoitettu tiedon siirtoon ja tallennukseen.

*www.cern.ch

**Suoraan lainattu teoksesta: Jaakohuhta Hannu, *IT-ensyklopedia*, Edita prima Oy, 2003

1 JOHDANTO

Nykyaikana melkein kaikelle on luotu persoonallinen imago tai brändi. Brändi rakentuu niin tuotteen yleisestä kuvasta kuin myös visuaalisesta ilmeestä. Vaikka tiedotteessa ei lukisikaan tuotteen nimeä, tuo punainen nauha mieleen Coca-Colan ja keltainen suorakulmainen kehys National Geographicin. Tällaisen imagon rakentaminen ei rajoitu pelkästään yritysmaailmaan, vaan monella instituutiolla ja tapahtumalla on nykyään oma visuaalinen ulkoasunsa. Ulkoasu on ennalta määrätty niin printatussa mediassa kuin myös sähköisessä mediassa. Verkkosivuja tarjoaville palveluille tämä luo haasteita. Verkkopalvelulla, joka tarjoaa tapahtuman järjestäjälle laajan valikoiman toiminnallisuuksia, mutta liian rajoittuneen sivujen ilmeen muokkauksen, päädytään helposti ns. kahden sivuston ongelmaan. Tällöin tapahtuman järjestäjät käyttävät verkkopalvelua sivuston toiminnallisuuksia varten ja toista, omaa kotisivua, visuaalisen ulkoasun esittämistä varten. Näin on käynyt myös CERNissä kehitetylle Indico- verkkoportaalille.

Indico- verkkoportaali on Python- skriptikielellä toteutettu, selainpohjainen tapahtumien hallinnointiohjelma. Indicoa on kehitetty vuodesta 2002 lähtien ja aikojen saatossa siitä on kasvanut laaja järjestelmä monine toiminnallisuuksineen. Kuitenkin palvelun käyttöliittymä, varsinkin suurempien tapahtumien (esim. konferenssien) osalta on jäänyt kehityksessä jälkeen. Käyttöliittymää on mainostettu alusta alkaen personoitavana, mutta se ei enää vastaa nykyisiä tapahtumien ja instituutioiden tarpeita.

Syntyi siis tarve käyttäjälähtöisten muutosten mahdollistamiseen. Tästä kehittyi osaprojekti, jonka toteutin heinä-syyskuussa 2008. Osaprojektini oli konferenssiosion käyttöliittymän parannustyöt. Tehtävänäni oli suunnitella ja toteuttaa sivuston rakenteelliset uudistukset sekä mahdollistaa käyttäjälähtöiset muutokset sivuston ulkoasuun. Ratkaisun myötä tehdyt toiminnallisuudet kirjoitettiin Python- skriptikieltä käyttäen. Python- kielen perusteet on rajattu tämän opinnäytetyön ulkopuolelle, koska työ painottuu käyttäjälähtöisten muutosten mahdollistamiseen, ja tehdyt ominaisuudet on toteutettavissa muitakin skriptikieliä käyttäen. Toteutunut projekti oli tämän insinöörityön kirjallisen osuuden perusta. Tehtyyn työhön palataan tämän insinöörityön lopussa pääluvussa 3, jossa esitellään varsinaisen työn lähtökohdat ja ongelmat, joihin etsittiin ratkaisumalleja, sekä lopulta toteutettu ratkaisu.

Tämän insinööri työn teoriaosuudessa vertaillaan erilaisia tekniikoita, joilla voidaan toteuttaa käyttäjälähtöistä personointia verkkosivuston ulkoasuun. Aluksi selostetaan lyhyesti verkkosivustojen rakentamisen perusteoria ja tämän jälkeen syvennyttään käytettävissä oleviin verkkosivuston ulkoasun personointikeinoihin. Työssä käydään kertauksen omaisesti läpi HTML- ja CSS-tekniikoiden perusteita. Tämän työn tarkoituksena ei kuitenkaan ole toimia aloitusoppaana HTML- ja CSS-tekniikoihin, vaan tarkoituksena on selvittää, kuinka HTML- ja CSS-tekniikoita voidaan soveltaa verkkosivuston ulkoasun luomisessa. Erityisesti, kuinka HTML- ja CSS- tekniikat soveltuvat ns. taulukottoman rakenteen ja sivuston käyttäjälähtöiseen ulkoasun muunnoksiin. Työn lukemisen helpottamiseksi kirjoittaja on pyrkinyt käyttämään suomenkielisiä versioita työssä esiintyvässä termistössä.

Loppupäätelmissä käydään läpi projektin lopputulokset ja mitä kaikkea jatkokehityksen kannalta tehty projekti mahdollistaa. Lisäksi loppupäätelmissä esitellään Indico versio 0.97b2, joka otettiin käyttöön 14.3.2010. Tässä uusimmassa versiossa on mukana tekemäni uudistukset ja muokkaukset tietyin muutoksin. Ne, mitkä ominaisuudet otettiin mukaan ja miten, selostetaan uuden version esittelyn ohella.

1.1 CERN

Toisen maailmansodan jälkimainingeissa tuli selväksi, ettei yksikään Euroopan valtioista pysty itseksensä hallitsemaan ydinfysiikan tutkimusta. Ainoaksi vaihtoehdoksi jäi usean valtion yhteistyö (joint effort). Tästä johtuen, lopulta vuonna 1952, CERN - Conseil Européen pour la Recherche Nucléaire (European Council for Nuclear Research) syntyi, ja otti tehtäväkseen Euroopan tutkimuskeskuksen suunnittelu- ja rakennustyöt. Vuonna 1954 rakennustyöt aloitettiin Genevessä, Sveitsissä. Tällöin myös organisaation nimi vaihtui ja vakiintui, kuitenkin akronyymi CERN säilyi. CERN on aikojen saatossa kasvanut maailman suurimmaksi hiukkasfysiikan tutkimuskeskukseksi, työllistäen 2600 henkilöä kokoajaisesti ja noin 6500 osa-aikaista työntekijää useissa tieteellisissä projekteissa.

CERN on tullut tunnetuksi tieteellisten saavutusten ja keksintöjen kotina. Näistä tunnetuimpia ovat:

- LHC (Large Hadronic Collider), suuri hiukkastörmäytin, joka käynnistettiin syyskuussa 2008 (fysiikka)
- LCG (LHC Computer Grid), suurin GRID-projekti, keskittynyt LHC:stä saadun datan prosessointiin, tallennukseen ja jakamiseen maailmanlaajuisesti (tietotekniikka)
- WWW (World Wide Web), jonka Sir Tim Berners-Lee kehitti 1989 (tietotekniikka)
- W ja Z hiukkasten löytäminen (fysiikka)

CERN on jaettu seitsemään laitokseen kuten IT(tietotekniikka). Jokainen laitos sisältää useita 'ryhmiä'. Ryhmät koostuvat osastoista. Tämä työ on toteutettu AVC (Audio, Visual and Conferencing Services) osastossa, UDS (User and Document Services) ryhmän alla.

1.2 Indico

InDiCo (Integrated Digital Conferencing) -projekti syntyi vuonna 2002 viiden organisaation ja instituutin yhteisprojektina. Projektin päätavoite oli luoda verkkopohjainen, monialustainen tapahtumien säilytys- ja hallinnointijärjestelmä. Järjestelmän oli tarkoitus mahdollistaa tapahtumiin liittyvien dokumenttien (esim. muistiinpanot, tiivistelmät, kalvot ja videot) ja metadatojen säilytyksen. Projekti kesti kaksi vuotta, jonka jälkeen CERN päätti jatkaa Indicon kehitystyötä. Myöhemmin Indico syrjäytti jo pahoin vanhentuneen CERN Agenda –tapahtumien hallinnointiohjelman. (CDS Indico Overview 2005, About Indico 2010, Ferreira 2006/2007)

Indico on nykyään intensiivisessä käytössä CERNin sisäisessä tapahtumien hallinnassa. Suurin osa organisaation sisäisistä tapahtumista hallinnoidaan Indicon kautta. Indicon välityksellä:

- Hoidetaan tilan- ja resurssien varauksia.
- Hallinnoidaan kokouksia, työpajoja, luentoja ja suurempia tapahtumia esim. konferensseja.
- Välitetään tarvittavia materiaaleja ja tietoja tapahtumaan osallistuvien kesken.

Indicoa käytetään myös CERNin ulkopuolella. Indicoa käyttää nykyään 40 eri organisaatiota tai instituuttia mm. USA:ssa sijaitseva hiukkastutkimuskeskus Fermilab. Monet isommista tapahtumista (pääosin konferenssit), jotka järjestetään CERNin ulkopuolella, hallinnoidaan Indicon kautta. (About Indico 2010.)



KUVA 1. Indicon virallinen logo

Indicon tarjoamia ominaisuuksia

Indico tarjoaa palveluita koko tapahtuman järjestämisen elinkaarelle:

- Tapahtumien kategoroinnin puumaiseen rakenteeseen tapahtumatyyppien ja osastojen mukaan.
- Automaattisen sivujen luonnin tapahtumille.
- Automaattiset ilmoitukset ja varoitukset, esimerkiksi ilmoitukset kokoukseen osallistuville paikan muuttumisesta.
- Palvelut konferenssien järjestämistä varten:
 - Rekisteröinti- lomakkeen muokkaus
 - On-line maksujen tuki
 - Tiivistelmien lähetys- ja arviointi.

Näiden perusominaisuuksien lisäksi Indico tarjoaa myös:

- Sisäänrakennetun tilanvarausjärjestelmän (tällä hetkellä käytössä vain CERNin sisäisesti)
- Sisäänrakennetun tuen videokokous-ohjelmistolle
- Monikielisen käyttöliittymän (tällä hetkellä kielinä ranska ja englanti)
- Tuen RSS-syötteelle
- Tuen eri aikavyöhykkeille
- Käyttöliittymän, jossa on otettu huomioon esteettömyys ja käytettävyys.
(About Indico 2010.)

2 TEORIA

2.1 Verkkosovelluksen perusteet

Tässä pääluvussa käydään läpi keskeiset asiat ja teknologiat, joita tarvitaan yksinkertaisen verkkosovelluksen luomiseen. Aluksi selostetaan verkkosovelluksen perusteet. Tämän jälkeen perehdytään HTML- ja CSS- kieliin. Aivan luvun loppuun tarkastellaan, kuinka HTML- ja CSS- tekniikoita hyödyntämällä voidaan vaikuttaa verkkosivun rakenteeseen ja ulkoasuun.

2.1.1 Verkko ympäristönä

Verkon toiminta perustuu perinteisesti asiakas-palvelin-malliin. Yleisesti se koostuu kolmesta osasta:

- asiakasohjelma
- palvelinohjelma
- yhteyskäytäntö eli protokolla.

Asiakasohjelma on usein käyttäjän käyttämä selainohjelma. Yleisimpiä selaimia ovat IE (Microsoftin Internet Explorer), Mozilla Firefox, Opera, Safari ja uusimpana Googlen Chrome. Tyypillisiä palvelimia ovat Apache, <http://httpd.apache.org/> ja IIS (Microsoftin Internet Information Server), <http://www.iis.net/overview>.

2.1.2 Kuinka asiakas-palvelin-malli toimii

Sovelluksen kaksi perusrakennetta ovat käyttöliittymä ja sovelluslogiikka.

Verkkosovelluksessa käyttöliittymä koostuu pääasiassa HTML-sivuista (erityisesti lomakkeista) sekä muista selaimen tarjoamista ominaisuuksista. Sovelluslogiikka rakentuu palvelimen tuottamista tekniikoista ja skriptikielistä.

Verkkosovellus koostuu palvelin – asiakas –yhteistyöstä. Palvelimella sijaitsee tietokannat ja sovelluslogiikka. Asiakaspäässä on kaikki mitä käyttäjä näkee ja voi käyttää. Joissakin tapauksissa osa sovelluslogiikasta on hyvä sisällyttää asiakaspäähän esim. AJAX-tekniikka. Tällöin voidaan käyttää asiakaspuolen skriptikieliä esim. Javascript. (Taulukko 1)

TAULUKKO 1. Verkkosovelluksen rakenne

Palvelintekniikat (Server)	Protokollaan liittyvät tekniikat	Asiakastekniikat (Client)
<ul style="list-style-type: none"> * CGI (mm. Perl, C, PHP, Sh) * Java Servletit * Upotetut tekniikat (SSI, PHP, ASP, JSP) * Sovelluspalvelimet 	<ul style="list-style-type: none"> URL:n rakenne ja * koodaus * Evästeet * Autentikointi menetelmät * Istunnon validointi 	<ul style="list-style-type: none"> * HTML * CSS * Javascript jne. * (DHTML) Javaohjelmat * (Appletit) * ActiveX, Hash, VRLM

Palvelimen ja –asiakkaan keskinäiseen tiedonsiirtoon käytetään HTTP (Hypertext Transfer Protocol) nimistä protokollaa. Verkkosovellus rakentuuakin näistä peräkkäisistä HTTP-siirtotapahtumista.

HTTP-siirtotapahtumassa palvelin kuuntelee, milloin asiakas tekee asiakaspyynnön. Kun pyyntö saapuu palvelimelle, käsittelee se saamansa pyynnön ja normaalisti palauttaa asiakkaalle vastaavan tiedon. Asiakaspyynnön mukana voi lähettää tietoa/parametrejä kolmea eri käytäntöä käyttäen:

- HTML-lomakkeelta POST-tai GET-metodia käyttäen.
- Hyperlinkin välityksellä, siihen liitetystä URL:n kyselyosasta GET-metodilla
- HTTP-otsakkeiden välityksellä (esim evästeet).

Palvelin puolestaan voi lähettää vastaustiedon kahdella eri tavalla:

- HTTP-otsakkeiden välityksellä
- Palvelimen vastauksen (HTTP-Response) dataosassa

2.1.3 Lyhyesti tilakoodista

Palvelin informoi asiakaspuolen sovellusta tilatiedoilla muun muassa sen, onnistuiko pyynnön suorittaminen vai ei. Tilatiedot koostuvat vastauskoodista ja 'selkeäkielisestä' tekstistä.

Vastauskoodit ovat jaettu seuraavanlaisesti:

- 100-199 Informatiiviset
- 200-299 Onnistuneet pyynnot
- 300-399 Uudelleen ohjatut
- 400-499 Virheelliset asiakkaan pyynnot
- 500-599 Palvelimen virhekoodit

Näistä koodista useinmiten normaali verkkokäyttäjä törmää koodiin *404 not found*, mikä tarkoittaa että haluttua sivua/tiedostoa ei löydy.

2.1.4 URL:n rakenne

WWW-dokumentit yksilöidään URL (Uniform Resource Locator) -osoitteiden avulla. URL syntyi vuonna 1994 Tim Berners-Leen vetämän työryhmän toimesta. URL:n yleinen kirjoitusasu on:

Käytettävä protokolla: protokollaan liittyvä osa

Protokollaan liittyvä osa on yleisesti muotoa:

//käyttäjä:salasana@palvelin:portti/URL-polku (Berners-Lee ym. 1994)

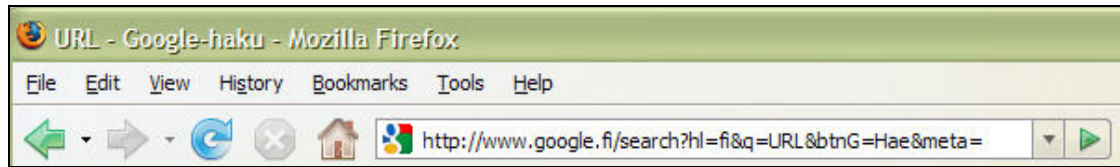
Tässä työssä URL:sta puhuttaessa käsitellään pääsääntöisesti HTTP-URL:ia.

HTTP-URL:ssa yleisen muodon osa: *käyttäjä:salasana@* ei ole sallittu ja URL-polku jakautuu seuraaviin valinnaisiin osiin:

- hakupolku
- kyselyosa

Hakupolku ja kyselyosa erotetaan kysymysmerkillä ja nimi-arvo- pareja erotetaan &-merkillä. Tämän vuoksi on huomioitava, että nimi-arvo- pareissa ei saa olla kysymys tai &-merkkejä. Jos näitä on, pitää ne koodata.

Esimerkiksi, kun kyseessä on google-hausta hakusanalla URL, on hakupolun osa tällöin: *www.google.fi/search*. Kyselyosa: *hl=fi&q=URL&btnG=Hae&meta=* koostuu neljästä parametri-parista, joista viimeisellä ei ole arvoa (Kuva 2).



KUVA 2: Esimerkki HTTP-URL:stä

2.2 HTML

HTML (HyperText Markup Language) on yksinkertainen sivunkuvauskieli, jolla WWW-sivujen rakenne kuvataan. HTML mahdollistaa sivun rakenteen jakamisen osioihin kuten otsakkeeseen, palstoihin, linkkeihin ja listoihin. HTML-koodi koostuu varsinaisesta tekstistä ja tekstin muodostusta ohjaavista koodimerkinnoista, ns. tageista, jotka on sijoitettu kärkisulkeis- merkkien (< >) sisään. Tagit ovat ennalta määritettyjä ja standardoituja.

2.2.1 HTML kehityshistoria

HTML syntyi vuonna 1989, kun CERNissä työskennellyt Tim Berners-Lee kirjoitti ensimmäisen ehdotelman Internetiä hyödyntävästä hyperteksti- järjestelmästä. Järjestelmän oli tarkoitus helpottaa CERNissä työskentelevien tiedemiesten välistä kommunikaatiota. Berners-Lee määrittä HTML:n ja kirjoitti ensimmäiset selain- ja palvelinohjelmistot vuoden 1990 lopussa. Samana vuonna työryhmään liittyi CERNin tietojärjestelmäinsinööri Robert Cailliau. Vaikka molemmat miehet olivat töissä CERNissä, ei HTML-projekti koskaan ollut CERNin virallistama.

2.2.1.1 Aiemmat versiot

Vuoden 1993 puolessa välin annettiin ehdotus ensimmäiseksi viralliseksi HTML-standardiksi. Tämä versio sisälsi 20 elementtiä, joista 13 on yhä käytössä (Tags used in HTML 1992). Kaksi vuotta myöhemmin julkaistiin HTML 2.0. Lisäyksenä

alkuperäiseen, tähän versioon oli lisätty elementit taulukkojen, lomakkeiden, kuvakarttojen esittämiseen.

Vuoden 1997 alussa HTML 3.2 julkaistiin W3C:n suosituksena. Tämä oli myös ensimmäinen täysin W3C:n standardoima versio IETF:n (Internet Engineering Task Force) lopetettua HTML-työryhmänsä loppuvuodesta 1996. HTML 3.2:sta poistettiin matemaattisten kaavojen esittämiseen tarkoitettut elementit ja lisättiin suurin osa Netscapen luomista ulkoasun kuvaamiseen tarkoitetuista elementeistä.

2.2.1.2 HTML 4

Alunperin HTML:n tarkoitus oli vain kuvata WWW-sivujen rakennetta, mutta käytön lisääntyttyä haluttiin vaikuttaa myös sivun ulkoasuun. Tähän tarpeeseen selainvalmistajat loivat HTML -määrittelyyn kuulumattomia elementtejä, jotka lisättiin lopulta myös HTML versioon 3.2. Kuitenkin loppuvuonna 1997, kun HTML 4.0 julkaistiin, korjattiin HTML 3.2:n virheet ja palattiin alkuperäiseen HTML:n tarkoitukseen. Se oli ensimmäinen selkeä askel erillisten tyylisivujen käyttöön, sen poistaessa suuren osan ulkoasun kuvaamiseen käytetyistä elementeistä.

Toinen suuri HTML 4.0 :n tuoma uudistus oli sen jako kolmeen eri variaatioon:

- Tiukka (strict), missä poistettujen elementtien käyttö on kielletty.
- Siirtymävaihe (transitional), missä poistetut elementit on hyväksytty
- Kehykset (frameset), missä suurin osa vain kehyksille (frame) tarkoitetuista elementeistä on hyväksytty.

Vuonna 1999 julkaistu HTML 4.01 sisälsi samat kolme variaatiota kuin HTML 4.0. Toukokuussa 2000 HTML 4.01 Strict- variaatioon pohjautuva HTML-standardi sai ISO standardin aseman. Vuonna 2008 HTML 4.01 ja ISO HTML olivat HTML:n uusimmat versiot.

2.2.2 Merkintä

HTML -merkintäkieli rakentuu elementeistä ja niiden määritteistä sekä sisällöstä. Elementit ovat ennalta määrättyjä HTML standardissa. Elementit alkavat kärkisulkeis-merkeillä erotetulla aloitustunnisteella (tagilla) ja päättyvät kautta -merkillä alkavaan lopetustunnisteeseen.

Merkintä: `Lihavoitu sisältö`

Tulos: **Lihavoitu sisältö**

HTML- elementeillä on kaksi perusominaisuutta: määritteet eli attribuutit (attribute) ja sisältö. Kuten elementit myös määritteet ovat ennalta määrättyjä HTML standardissa. Elementtiin voi liittyä yksi tai useampi määrite. Määritteet kirjoitetaan elementin alkutagin sisään. Aluksi merkitään määritteen nimi ja tämän jälkeen arvo yhtäkuin-merkin jälkeen lainausmerkeissä.(Korpela & Linjama 2006, 76.)

``

Sisältö on vapaasti kirjoitettavaa tekstiä, joka näkyy sivulla itse sisältönä. Poikkeuksen sisällön kirjoittamiseen tuo erikoismerkit, jotka on kirjoitettava koodimuodossa &-merkin (ampersand) ja puolipisteen väliin (esimerkiksi sitova välilyönti, joka kirjoitetaan muodossa ` `).

Elementit

HTML- elementit voidaan karkeasti jakaa kahteen ryhmään: rakenteellisiin ja esityksellisiin. Rakenteelliset elementit kuvaavat tekstin tarkoitusta ja sivun rakennetta. Näitä ovat mm. otsikko- elementit, *H1-H6* ja taulukko- elementti, *table*. Esitykselliset elementit kuvaavat tekstin ja sivun ulkoasua. Tällaisia ovat mm. lihavointi- elementti *b* ja kirjasinlajin määräävä elementti *font*. Suurin osa näistä esityksellisistä elementeistä on poistettu käytöstä HTML 4 standardin ja CSS:n myötä, eikä niitä tulisi enää käyttää. Niitä esityksellisiä elementtejä, jotka jäivät HTML 4-standardiin, hyödynnetään mm. sivustojen puheselaimissa.

Yliselementit

HTML-kielessä on kaksi yleiselementtiä: *div* ja *span*. Näitä käytetään ”kääreinä”, kun halutaan erottaa jokin dokumentin osa omaksi kokonaisuudekseen. Yleissääntönä on, että *div* on lohkotason elementti ja *span* sisätason. Tämä tarkoittaa, että *div* voi sisältää muita lohko- ja sisäelementtejä, kun taas *span* vain tekstiä ja sisäelementtejä. (Korpela & Linjama 2006, 80.) Yleisesti *div*-elementtiä käytetään sivun taiton rakentamisessa. Tästä lisää luvussa 2.4.3.

HTML-dokumentti

HTML- dokumentti on aina perusmuotoa:

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
"http://www.w3.org/TR/html4/loose.dtd">
<html>
  <head>
    <meta http-equiv="Content-Type"
    content="text/html; charset=iso-8859-1">
    <title> . . . </title>
  </head>
  <body>
    tänne tulee itse rakenne ja sisältö
  </body>
</html>
```

Alussa olevalla *doctype*-määrittelyllä on kaksi tarkoitusta:

- Se helpottaa sivun merkkauksen tarkistamista, kun käytetään tarkistusohjelmia eli validaattoreita.
- Se auttaa joitakin selaimia toimimaan paremmin HTML-standardin mukaisesti.

Doctype –määritteen jälkeen alkaa varsinainen HTML-dokumentti. Koko dokumentti kirjoitetaan `<html>`- elementin sisään. `<html>`- elementti on jaettu kahteen pakolliseen osaan: otsake-elementtiin `<head>` ja runko-elementtiin `<body>`. (Korpela & Linjama 2006, 81.)

Otsake-elementti koostuu pakollisesta *title*-elementistä, joka nimeää dokumentin ja näkyy yleensä selainikkunan yläpalkissa. Lisäksi otsake-elementin sisään voidaan kirjoittaa muita elementtejä, kuten *meta*-elementti. Nämä elementit eivät ole sivun varsinaista sisältöä, vaan ne kuvailevat sitä eri tavoin tai liittyvät siihen. (Korpela & Linjama 2006, 82.) Mainittakoon myös, että CSS-tyylitiedostoja käytettäessä viittaus CSS-dokumenttiin kirjoitetaan otsake-elementin sisään. Tästä kerrotaan lisää luvussa 2.3.

Dokumentin runko-osan sisään tulee sivun varsinainen rakenne ja sisältö, eli se mitä käyttäjä näkee selainikkunassaan. Koko runko-osa kirjoitetaan `<body>` -elementin sisään. Dokumenttia tallennettaessa tiedostonpääteenä käytetään `.htm` tai `.html`-päätetä.

2.2.3 Staattinen verkkodokumentti

Verkkodokumentteja kuvataan HTML-kielellä. Perinteisesti verkkodokumentit ovat tehty staattisiksi eli muuttumattomiksi. Tämä tarkoittaa, että sivujen sisältöä ja ulkoasua voi muuttaa vain sivun tekijä tai ylläpitäjä. Sivuston käyttäjä ei siis voi tehdä muutoksia. Tämä rajoittaa vuorovaikutusmahdollisuuksia.

Staattiset verkkodokumentit sopivat mainiosti tilanteisiin, joissa dokumentteja täytyy päivittää harvoin ja vuorovaikutusmahdollisuuksia ei tarvita. Kuitenkin on tilanteita, joihin staattiset verkkosivut eivät riitä. Tästä lisää seuraavassa luvussa.

2.2.4 Dynaamiset verkkosivut

Dynaamiset verkkodokumentit sisältävät mahdollisesti muuttuvaa tietoa. Joskus sivujen tietojen haku on asiakaslähtöistä (esimerkiksi verkkohaku) ja joskus tieto voi olla alati muuttuvaa (uutispalvelut, huutokaupat, sääpalvelut ja muut reaaliaikapalvelut). Näihin tarpeisiin on kehitetty tekniikoita, joilla verkkosivut voidaan luoda osittain tai kokonaan dynaamisiksi. Lähes poikkeuksetta tähän tarvitaan jonkin muotoista ohjelmointia. Jos dynaaminen sivusto on rakennettu jotakin tiettyä sovellusaluetta ajatellen, voidaan puhua verkkosovelluksesta. Koska verkkosovellukset toimivat pääsääntöisesti käyttäjälähtöisesti, puhutaan myös interaktiivisista verkkosivuista.

2.3 CSS-tyylisivut

CSS on kieli, jolla kerrotaan selaimille sääntöjen avulla, millainen ulkoasu tietyllä sivulla on. Tällaista sääntökirjastoa kutsutaan tyyliohjeeksi. Tiedostoa jossa on sivua koskeva tyyliohje kutsutaan tyylitiedostoksi tai CSS-tiedostoksi. Tyyliohjeita voi soveltaa monissa tapauksissa, esim. XML:n (Extensible Markup Language) parissa. Tässä työssä keskitytään ainoastaan CSS:n käyttöön HTML-muotoisten dokumenttien kanssa.

Ennen CSS:n kehitystä ja varhaisten CSS-versioiden aikana nämä samat tyyliohjeet laadittiin HTML-dokumenttien sisään. Tämä on toimiva tapa yhä, kun kyseessä on muuttumaton, yksinkertainen sivu. Monimutkaisempien ja varsinkin ulkoasultaan muutettavien sivustojen kohdalla tyylien ylläpitäminen HTML-koodin yhteydessä on turhan työlästä. On huomioitava, että jos sivusto koostuu useasta sivusta, tulee HTML-koodin yhteyteen kirjoitetut tyylit kopioida jokaiseen HTML-tiedostoon. Tämä aiheuttaa HTML-tiedostojen koon kasvamista ja ylläpidon hankaloitumista, sillä kaikki tulevat muutokset on muistettava tehdä jokaiseen tiedostoon erikseen. Tyylisivuja käytetään sivuston elementtien ulkoasun määrittelyyn. Tyylisivuilla dokumentin rakenteen ja sisällön voi erottaa toisistaan. Kun ulkoasu ja rakenne erotetaan omiksi kokonaisuuksiksi, voidaan ulkoasuun tehdä muutoksia koskematta rakenteeseen tai sisältöön.

Tyyliohje voidaan määrittää HTML-dokumentin alussa otsake-osiossa (*head*), jolloin se koskee vain kyseessä olevaa HTML-dokumenttia. Tai se voidaan määrittää ulkoisesti, erillisessä tiedostossa, jolloin sen määrittelemät tyyliohjeet koskevat kaikkia niitä HTML-dokumentteja, joissa tähän erilliseen tiedostoon viitataan.

Erillisestä tyylitiedostosta hyötyvät mm. yritykset, jolloin sivuston ulkoasu voidaan ”standardoida” yrityksen imagon mukaiseksi. Erillisestä tyylitiedostosta on hyötyä myös, kun halutaan antaa käyttäjälle mahdollisuus valita erilaisten ulkoasujen väliltä (sivuston personointi). Samoin kuin useissa julkaisuohjelmissa (esim. Adoben InDesign), voidaan CSS:ssä määrittää elementtien paikka käyttämällä X- ja Y-koordinaatteja. Tästä lisää luvussa 2.4.1, asemointi.

Parhaimmillaan CSS-tyylitiedostojen käytöllä voidaan luoda paikkatietoja dynaamisesti vaihtavia ohjelmia, jolloin sivulla oleville elementeille voidaan antaa *drag and drop*-ominaisuus. Tämä tarkoittaa, että sivuston käyttäjä voi vaihtaa sivuston elementtien paikkaa haluamakseen. Tästä on hyvänä esimerkkinä iGooglen etusivu. Sivun rakenteen lisäksi tyyliohjeella voidaan toteuttaa tekstinkäsittelyohjelmista (esim. Microsoftin Word) tuttuja tyylejä. Tyylejä voi määritellä eri otsikkotasolle, kuvateksteille, mainosteksteille ja leipätekstille. Tyylejä käyttämällä samat kirjaisinlajin valintaan, kokoon, väriin ja tehosteisiin vaikuttavat määrittelyt siirtyvät kaikkiin kyseessä olevaa tyyliä käyttäviin tekstiosiin.

2.3.1 CSS:n kehityshistoria

Jo HTML:n kehityksen alkuvaiheissa painotettiin, että HTML:n rinnalle tulee erillinen tyylimäärittely. Mutta koska tyyliohjeiden määrittely eteni hitaasti, selainten valmistajat alkoivat lisäämään HTML:ään tyyliohjeita mukailevia piirteitä, kuten nyt jo vanhentunut *font*-tagi. Sivujen laatijat oppivat ja tottuivat käyttämään näitä HTML:n sisäisiä tyyliohjeita. Osaltaan myös siksi, että varhaiset graafiset selaimet esittivät sivut hyvin yksinkertaisina, eikä monimutkaisempaa tyyliohjeistusta tarvittu.

Hyvä esimerkki CSS:n kehityksen hitaudesta on, kun CSS 1 suositus julkaistiin 17.12.1996 oli IE edennyt jo kolmanteen versioonsa.

CSS versiot

CSS 1:stä tuli ensimmäinen W3C:n hyväksymä CSS suositus vuoden 1996 joulukuussa. Sen ominaisuuksia olivat mm.

- eri elementtien värien määrittely
- eri elementtien sijoittelun määrittely
- marginaalin, reunaviivan, täytön määrittelyt
- määreiden yksilöllinen identifiointi ja luokittelu

Vuonna 1998 ilmestyi CSS 2, joka toi CSS 1:ssä esiintyneiden ominaisuuksien lisäksi mahdollisuuden elementtien sijoitukseen kolmella eri tavalla:

- Absoluuttinen sijoittelu (Absolute)
- Suhteellinen sijoittelu (Relative)
- Kiinteä sijoittelu (Fixed)

CSS 2 sisälsi myös tuen laitteistokohtaisille tyylisivuille ja näin ollen mahdollisti visuaalisten ja audittiivisten selainten käytön. Muita uusia ominaisuuksia olivat mm. tuki kielen lokalisaatioon ja tekstin numerointiin. (Cascading Style Sheets... 2008)

CSS 2:sta julkaistiin parannettu versio CSS 2.1, joka korjasi CSS 2:ssa esiintyneitä virheitä ja poisti heikosti tuettuja ominaisuuksia. CSS 2.1:stä ei koskaan tullut W3C:n hyväksymää suositusta. CSS 2.1 on kuitenkin käytössä ns. *de facto* –standardina (Korpela 2008, 56). Vuonna 1999 aloitti W3C uuden CSS2 määrittelyn kehittämisen. Tämä uusi CSS 2.2 määrittely sisältää mm. asemointiin käytettävät määreet sekä dokumenttien tulostamiseen vaikuttavia tyylimääreitä.

2.3.2 CSS tyylien käyttö

Kuten luvussa 2.3 mainittiin, tyyliohje voidaan laatia erilliseksi tiedostoksi tai upottaa HTML-dokumenttiin. HTML-dokumenttiin upottamisessa on vielä kaksi vaihtoehtoa. Tyyliohje voidaan laatia HTML-dokumentin *head*-osioon, jolloin tyyliohjeen määreet koskevat koko dokumenttia. Toinen tapa on, että tyyliohjeet voidaan upottaa yksittäiseen HTML-elementtiin, esimerkiksi otsikkoelementin *H3* sisälle style-attribuuttia käyttäen. Jälkimmäistä tapaa ei kuitenkaan suositella, sillä silloin palataan samaan tilanteeseen missä oltiin ennen CSS-tyylien määrittelyä. Yksittäiseen HTML-elementtiin upotetut tyyliohjeet ovat käytännöllisiä poikkeustilanteissa, kun näitä elementtejä on HTML-dokumentissa vain vähän ja kun halutaan esimerkiksi yliajaa jokin ylemmän tason tyyliohje.

2.3.3 CSS –merkintä

CSS-tyyliohjeen syntaksi on suhteellisen helppo oppia, koska sen perussäännöt säilyvät samoina tyyliohjeen sijainnista ja versiosta huolimatta. CSS-tyyliohjeessa aluksi nimetään valitsin. Valitsimista lisää luvussa 2.3.4. Valitsinta koskevat ominaisuudet erotetaan toisistaan puolipisteellä ja ominaisuuksien arvot kaksoispisteellä. Selkeyden vuoksi erillisessä tyyliohjetiedostossa eri ominaisuudet on syytä erottaa toisistaan välilyönnein.

Esimerkiksi kun halutaan määrittää erillisellä tyyliohjeella otsikkotasolle H2 kirjasinkoko 20pt ja väri vihreä, on merkintä seuraavanlainen:

```
H2 {
    color:green;
    font-size:20pt;
}
```

Kun samanlaiset määritykset upotetaan HTML-elementin sisälle, on merkintä seuraavanlainen:

```
<H2 style="color:green;font-size:20pt;">
```

2.3.4 CSS -valitsimet

CSS2:n määrittelyyn on lisätty joukko uusia valitsimia, joiden avulla tyyliohjeet voidaan linkittää HTML-elementteihin.

- Universaali valitsin `*` vaikuttaa kaikkiin dokumentin elementteihin
- Tyyppivalitsin eli HTML-elementin nimi (esim. `table`) määrittää yksittäisen elementin muotoilun

CSS tyylimääreet voidaan asettaa valmiille HTML-elementeille, kuten otsikkotasot *H1-H6*, palstalle *p* ja linkkielementille *a*. Mutta määreet voidaan ottaa käyttöön myös itse identifioituille kokonaisuuksille. Tällöin määreen linkittämiseen tiettyyn elementtiin käytetään CSS valitsimia (CSS selector).

Kaksi yleisintä valitsinta ovat: Luokka-attribuutti ja id-attribuutti.

Luokka-attribuutti (`class`), merkitään HTML-dokumentissa seuraavanlaisesti:

```
<HTML-elementti class="luokan_nimi">
```

Tyyliohjeessa luokka-attribuutteihin viitataan etupisteellä luokannimen edessä: `.luokan_nimi`. Juuri tätä luokka-attribuuttia käytetään apuna tyyliohjeen periytymisessä.

Id-attribuutti merkitään HTML-dokumentissa samoin kuin luokka-attribuutti:

```
<HTML-elementti id="id_nimi">
```

Tyyliohjeessa id-attribuuttiin viitataan # -merkillä id:n nimen edessä:

```
#id_nimi
```

On huomioitava, että tietyllä id-attribuutin arvolla esiintyvä elementti saa esiintyä vain kerran HTML-dokumentissa. Luokka-attribuuttien käyttöä ei ole rajoitettu.

Näiden edellämäinittujen valitsimien lisäksi on olemassa joukko valitsimia, jotka ovat harvenaisempia käytössä, mutta voivat olla suureksi hyödyksi. Näitä ovat:

Jälkeläisvalitsin A B (esim. H1 strong)

- Määrittää elementti B:n muotoilun, jos se on elementti A:n jälkeläinen (lapsi, lapsenlapsi jne.)

Lapsivalitsin A>B (esim H1 > strong)

- Määrittää elementti B:n muotoilun, jos se on elementti A:n lapsi.

Sisarusvalitsin A+B (esim H1+H2)

- Määrittää elementti B:n muotoilun, jos se on elementti A:n sisar, eli elementit ovat samalla hierarkisella tasolla.

2.3.5 Kaskadisuus ja periytyvyys

CSS tarkoittaa ”putousmaista” mallia. Tämä tarkoittaa, että tyylien määrittelyssä otetaan huomioon aiemmat määreet huomioon ja määreissä laskeudutaan yleisestä tasosta yksityiseen tasoon. Kaskadisäännöt ratkaisevat, kumman tyyliohjeen määre on elementin muotoileva tyyli, kun kahdessa eri tyyliohjeessa määritetään saman elementin sama ominaisuus. (Korpela 2008, 496.)

CSS -tyylejä voidaan myös periyttää. Jos elementin jokin ominaisuus on määritelty periytyväksi, eikä sille ole asetettu mitään tiettyä arvoa, perii se ympäröivän elementin ko. ominaisuuden arvon (Korpela 2008, 513). Esimerkiksi tyyli *p.mainosteksti* perii kaikki HTML-elementille *p* luodut määrittelyt.

2.3.6 Erillinen (ulkoinen) tyylitiedosto

Yksi etu ulkoisissa tyylitiedostoissa on se, että niistä sivuston laatija voi luoda omia tyylikirjastoja, josta ne voidaan ottaa käyttöön tarvittaessa.

Ulkoinen tyylitiedosto otetaan käyttöön HTML-dokumentissa seuraavanlaisesti:

```
<head>
  <link rel="stylesheet" type="text/CSS"
    href="tyylisivun_polku/tyylisivu.CSS"/>
</head>
```

Useampia tyylisivuja voidaan ottaa myös käyttöön samanaikaisesti. Tällöin käytetään import-lauseita:

```
<head>
  <style>
    @import url("tyylisivu.CSS")
    /*mahdolliset sisäiset tyyliohjeet tulevat import
    lauseen jälkeen*/
  </style>
</head>
```

Import-lauseella voidaan myös liittää tyylitiedostoja tyylitiedoston sisään, jolloin import-lause on kirjoitettava tiedoston alussa.

2.4 Verkkosivun rakenteen luominen

Seuraavassa keskitytään verkkosivun rakenteiden luomiseen ja siihen kuinka HTML-elementit voidaan asemoida sivulle. Lisäksi selostetaan kaksi eri tapaa tuottaa sivun taitto ja vertaillaan näitä taittotapoja keskenään.

2.4.1 Asemointi

Asemoinnilla määritetään elementtien sijoitus sivulla. Mikäli erillistä asemointia ei määritetä, selain lataa elementit omien sääntöjen mukaan. Usein siis siten, että erilliset lohkoelementit sijoittuvat allekkain.

HTML:ssä dokumentin elementtien sijoitukseen ei vaikuta toiset samassa dokumentissa olevat elementit, vaan elementti sijoittuu usein seuraavaan vapaaseen paikkaan. HTML-elementtejä on voitu sijoitella käyttämällä align- ja valign-attribuutteja, mutta nämä ovat vain ohjeellisia sijoitteluita, joita selain ei ole pakko toteuttaa.

Tämän takia ainoa järkevä tapa asemoida elementti, ennen CSS 2 määrittelyä, oli käyttää HTML-taulukkoelementtejä. Tästä lisää luvussa 2.4.2, taulukkotaitto.

CSS 2:n mahdollistaessa elementtien asemoinnin tyyliohjeella (CSS positioning), elementtien paikan määrittely sekä selkiytyi että tarkentui. Elementtejä pystytään jopa asettamaan päällekkäin, mikä ei ollut aiemmin mahdollista.

Sijoittelun apuna ovat ominaisuudet *top*, *bottom*, *left* ja *right* ja jokainen näistä voi saada arvona joko lukuarvon tai prosenttiluvun. On huomioitava, että sijoitteluominaisuudet eivät periydy kuten useimmat CSS-ominaisuudet. (Peltomäki 2006, 266.)

Sijoitteluun on käytettävissä erilaisia tapoja:

- Normaali (Normal)
- Absoluuttinen (Absolute)
- Suhteellinen (Relative)
- Kiinteä (Fixed)

On huomioitava, että elementin sijaintia kuvaruudulla ei voi määrätä. Esitysalue on selainikkuna, jonka kokoon ja sijaintiin ei CSS:llä voi vaikuttaa. Lisäksi siihen miten elementit asettuvat ja näkyvät käyttäjälle, vaikuttavat käytettävän näyttölaitteen koko ja resoluutio.

CSS:ssä asemointi on siis aina suhteellista, vaikka käytetään termejä absoluuttinen ja suhteellinen asemointi. Mahdollisimman lähelle absoluuttista asemointia päästään, kun elementti määritetään sivun vasempaan yläkulmaan, joka on sivuston peruskoordinaatiston 0-kohta.

Oletusarvona sijoittelussa on normaali asemointi, joka tarkoittaa, että elementti sijoitetaan kuten HTML:ssä tavallisesti ilman erillisiä tyyliohjeita. Absoluuttisesta ja suhteellisesta sijoittelusta kerrotaan lisää seuraavassa luvussa.

Absoluuttisen ja suhteellisen asemoinnin lisäksi on olemassa niin kutsuttu kelluva asemointi (liquid design). Kelluvassa asemoinnissa käytetään apuna CSS-määrettä *float*. Tästä lisää luvussa 2.4.4.

Absoluuttinen ja suhteellinen sijoittelu

Koska jokaisen elementin oletusarvoisena sijoitteluna käytetään kiinteää normal-sijoittelua, on sen uusi paikka määriteltävä joko absoluuttiseksi tai suhteelliseksi ennen muita tyylimäärittelyjä. Absoluuttiseksi määritetty elementti voi tulla toisen elementin päälle. Tästä voi olla hyötyä, mutta on myös varottava virheellisten sijoittelujen aiheuttamaa elementtien limittäisyyttä.

On myös huomioitava, että absoluuttiseksi asetetun elementin sijoitus lasketaan suhteessa dokumentin koordinaatteihin, eikä selainikkunan koordinaatteihin. Tämä tarkoittaa sitä, että elementin paikka säilyy paikallaan, vaikka käytettäisiin sivun vierityspalkkia.

Absoluuttista sijoittelua käytetään ensisijaisesti animaatioiden, pelien ja multimediaosuuksien toteutuksessa. Esimerkiksi sillä voi asettaa muuttuvan mainoslauseen kuvan päälle ilman, että täytyisi luoda erillisiä tekstillisiä kuvia.

Muihin käyttötarkoituksiin ja varsinkin tekstien asemointiin kannattaa käyttää suhteellista asemointia (relative). Suhteellisessa sijoittelussa elementin nollakohta on sen normaalissa sijoittelussa esiintyvä kohta. Joten määrittelyllä `top:15px` elementti

siirtyy normaalikohdastaan 15 pikselin leveyttä alaspäin. Vastaavasti negatiivisella arvolla elementin saa siirrettyä ylöspäin.

Kiinteä sijoittelu (fixed)

CSS 2 -määrittely toi mukanaan myös mukanaan kiinteän asemoinnin. Kiinteässä asemoinnissa elementin sijainti lasketaan suhteessa selainikkunan koordinaatteihin. Tällöin voidaan asettaa elementti kiinteästi siten, että sen paikka selainikkunassa säilyy, vaikka sivun vierityspalkkia käytetään. Tätä on käytetty hyödyksi, esimerkiksi kiinteän menu-valikon luomisessa, kun HTML-dokumentti on liian pitkä yhdellä sivulla näytettäväksi.

Kiinteässä sijoittelussa on se haittapuoli, että vanhemmat selainversiot ja IE6 eivät tue tätä sijoitustapaa (Peltomäki 2006, 269). Joten mikäli tätä sijoitustapaa käyttää, tulee sen toimivuus testata usealla eri selaimella ja selainversiolla, tai luotava esimerkiksi Javascriptiä käyttämällä vaihtoehtoisia ratkaisuja niille selaimille/selainversioille, joilla tämä sijoitustapa ei toimi.

2.4.2 Taulukkotaitto

Koska CSS:n määrittelyn valmistuminen viivästyi suhteessa sivuston laatijoiden tarpeisiin, oppivat laatijat käyttämään taulukoiden luomiseen tarkoitettuja HTML-elementtejä myös sivun taiton laatimiseen. Osan mielestä CSS:n käyttö asemoinnissa on yhä liian monimutkaista ja epäluotettavaa, ja siksi he käyttävät yhä taulukkotaittoa, vaikka muuten olisivat jo siirtyneet CSS:n käyttöön.

Taulukkotaitossa sivun rakenne hajoitetaan yhdeksi tai useammaksi sisäkkäiseksi taulukoksi siten, että taulukon solut muodostavat sivun rakennekokonaisuuksia. Taulukon, taulukon sarakkeiden ja solujen muodostamiseen käytetään vastaavia HTML-elementtejä; *table*, *tr*, *td*. Esimerkiksi laadittaessa sivu (Kuva 3), jossa on otsikolle on varattu oma tila sivun yläosaan ja loppu sivu on jaettu kahtia, voidaan toteuttaa esimerkin 1 mukaisella HTML-koodilla.

```

<html>
  <head></head>
  <body>
    <table width="100%" height="100%" cellspacing="0">
      <tr valign="top" height="50%">
        <td width="100%" colspan="2"
          bgcolor=#666666>
          Tässä on otsikko
        </td>
      </tr>
      <tr height="50%">
        <td width="50%" bgcolor=#999999>
          Tässä on vasemmanpuoleinen palsta
        </td>
        <td width="50%" bgcolor=#dddddd>
          Tässä on oikeammanpuoleinen palsta
        </td>
      </tr>
    </table>
  </body>
</html>

```

ESIMERKKI 1. Taulukolla tuotettu taitto

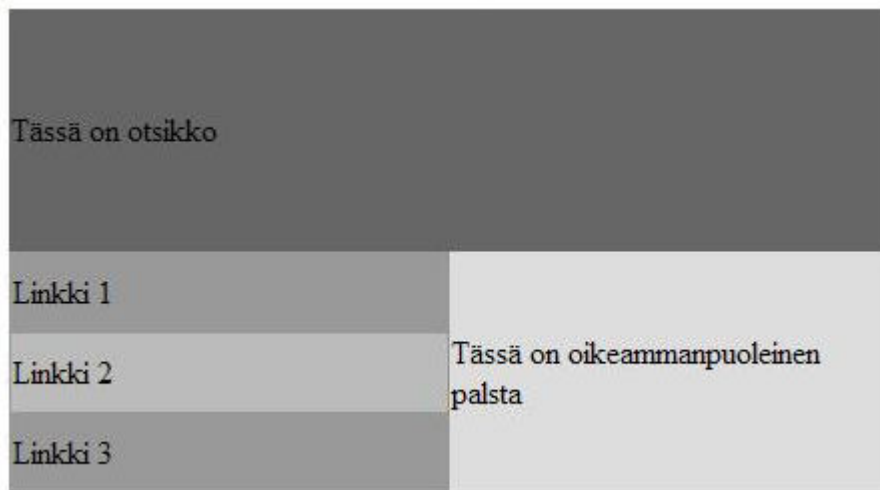
Tässä on otsikko	
Tässä on vasemmanpuoleinen palsta	Tässä on oikeammanpuoleinen palsta

KUVA 3. Taulukolla tuotettu taitto. Selkeyden vuoksi taustat on määritelty eri harmaan sävyille

Sivun rakenne voi myös sisältää sisäkkäisiä taulukkoja. Esimerkiksi jos halutaan tehdä menu-valikko. Tällöin yksittäinen solu jaetaan osiin uudella taulukolla (Esimerkki 2)(Kuva 4).

```
<html>
  <head></head>
  <body>
    <table width="100%" height="100%" cellspacing="0">
      <tr valign="top" height="50%">
        <td width="100%" colspan="2" bgcolor=#666666>
          Tässä on otsikko</td>
        </tr>
      <tr height="50%">
        <td width="50%" bgcolor=#999999>
          <table width="100%" height="100%"
            cellspacing="0">
            <tr>
              <td bgcolor=#999999>
                Linkki 1</td>
            </tr>
            <tr>
              <td bgcolor=#bbbbbb>
                Linkki 2</td>
            </tr>
            <tr>
              <td bgcolor=#999999>
                Linkki 3</td>
            </tr>
          </table>
        </td>
        <td width="50%" bgcolor=#dddddd>
          Tässä on oikeammanpuoleinen palsta </td>
        </tr>
      </table>
    </body>
  </html>
```

ESIMERKKI 2. Sisäkkäiset taulukot.



KUVA 4. Sisäkkäiset taulukot

Kuten kuvia ja niiden koodeja vertailemalla voidaan huomata, mitä monimutkaisemmaksi sivun rakenne tulee, sitä monimutkaisemmaksi HTML-koodikin kasvaa. Sivun rakenne voidaan myös laatia käyttämällä CSS-tyylitiedostoa. Tästä lisää seuraavassa luvussa.

2.4.3 Taulukoton taitto (tableless design)

Taulukottomassa taitossa HTML:n *table*-taulukkoelementti korvataan *div*-elementeillä ja niiden muotoilu toteutetaan CSS-tyyliohjeella. Jokaisella eri ominaisuudet omaavalla *div*-elementillä on oma yksilöllinen id-tunnus. Esimerkiksi kun laaditaan luvussa 2.4.2 esitetyn esimerkin mukainen sivu käyttämällä CSS:ää apuna, on tuloksena seuraavanlainen koodi:

```
<html>
  <head>
    <style type="text/CSS">
      #otsikko, #osa1, #osa2 {
        position:absolute;
        height:50%;
      }

      #otsikko{
        top:0;
        left:0;
        width:100%;
        background:#666666;}
    </style>
  </head>
  <body>
```

```

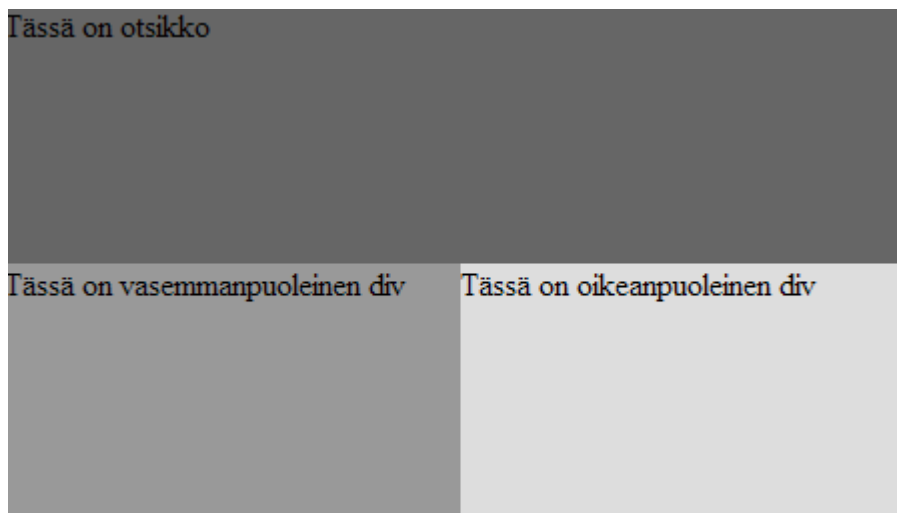
        #osa1, #osa2 {
        top:50%;
        width:50%;
        }

        #osa1 {
        left:0;
        background:#999999;
        }

        #osa2 {
        left:50%;
        background:#dddddd;
        }
    </style>
</head>
<body>
    <div id="otsikko">Tässä on otsikko</div>
    <div id="osa1">Tässä on vasemmanpuoleinen div</div>
    <div id="osa2">Tässä on oikeanpuoleinen div</div>
</body>
</html>

```

Huomattava on, että yksinkertaistamisen nimissä on tyyliohje upotettu HTML-dokumentin *style*-elementtiin.



KUVA 5. Esimerkin mukainen sivu

Huomataan, että CSS-tyyliohjetta käytettäessä sivun HTML- osio on huomattavasti yksinkertaisempi ja lyhyempi.

Vastaavasti monimutkaisempi rakenne, jossa vasemmanpuoleinen sarake on korvattu 'linkeillä', on CSS:ää käyttämällä seuraavan näköinen:

```
<html>
  <head>
    <style type="text/CSS">
      #otsikko, #osa1, #osa2 {
        position:absolute;
        height:50%;
      }

      #otsikko{
        top:0;
        left:0;
        width:100%;
        background:#666666;
      }

      #osa1, #osa2 {
        top:50%;
        width:50%;
      }

      #osa1 {
        left:0;
      }

      #osa2 {
        left:50%;
        background:#dddddd;
      }

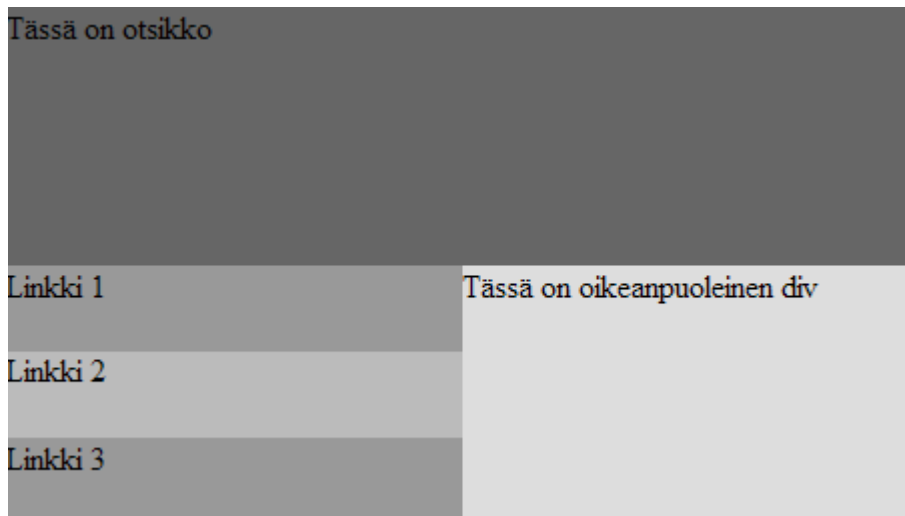
      .linkki1 {
        height:33%;
        background:#999999;
      }

      .linkki2 {
        height:34%;
        background:#bbbbbb;
      }
```

```

        </style>
    </head>
    <body>
        <div id="otsikko">Tässä on otsikko</div>
        <div id="osa1">
            <div class="linkki1">Linkki 1</div>
            <div class="linkki2">Linkki 2</div>
            <div class="linkki1">Linkki 3</div>
        </div>
        <div id="osa2">Tässä on oikeanpuoleinen div</div>
    </body>
</html>

```



KUVA 6. Esimerkin mukainen sivu

Tässä sisäkkäiset taulukkorakenteet on korvattu *div*-elementeillä siten, että ylemmällä tasolla olevat *div*:it ovat ensimmäisen taulukon soluja. Ja sisemmän taulukon solut ovat seuraavan tason *div*:ejä, jotka ovat siis yhden erillisen *div*:n sisällä. Koska muotoilun osalta linkit 1 ja 3 ovat samanlaisia, voidaan ne luokitella saman *class*-tunnuksen alle ja siten käyttää samaa tyyliohjetta.

Huomataan, että mitä monimutkaisempi sivuston rakenne on, sitä yksinkertaisempi *div*-rakenne on perinteiseen taulukkotaittoon verrattaessa. Koska muotoilut määritetään erillisellä tyyliohjeella, voidaan sivun rakenne muokata HTML-koodiin koskematta. Tämä on yksi taulukottoman rakenteen vahvimpia ominaisuuksia. Lisää vertailua taulukottoman ja taulukkotaiton välillä on luvussa 2.4.5, jossa vertaillaan taulukkotaiton ja taulukottoman rakenteen välisiä hyviä ja huonoja puolia.

2.4.4 Kelluva rakenne

Taulukottoman rakenteen käyttö vaatii kelluvaa asemointia eli ns. float-määrettä. Kun sivun rakenteellinen osa määritetään kelluvaksi, kelluu se irrallaan kuin veden päällä ja float:n attribuutit *right* ja *left* määräävät kumpaan reunaan rakenneosa ajautuu. On huomioitava, että kelluvasta elementistä tulee lohkotyyppinen elementti. Kun halutaan, että kelluvan elementin jälkeinen elementti ei siirry kelluvan viereen vaan alle, on viimeistä kelluvaa elementtiä seuraavalle elementille asetettava *clear*-määre. *Clear*:n attribuutit; *both*, *left* ja *right*, määräävät kummalla puolella elementtiä kelluvat elementit eivät ole sallittu. Usein kannattaa käyttää *both*-attribuuttia yhden sivun määräämisen sijaan, koska tällöin ei tarvitse muistella kuinka edelliset elementit ovat määritelty. (Visual formatting model 2009.)

2.4.5 Taulutaitto vs. taulukoton taitto

Nykyään on olemassa kolme rakenteellista vaihtoehtoa luoda verkkosivun rakenne:

- käyttämällä div-rakenteita
- käyttämällä taulukkorakennetta
- käyttämällä kahden edellisen vaihtoehdon hybridiä

Verkkosivujen kehittäjien parissa tämä *div* vastaan taulukot on kuuma puheenaihe. Tähän kysymykseen ei ole olemassa yhtä oikeaa vastausta, vaan se mikä rakennevaihtoehto on oikea, määräytyy sivuston tarpeiden mukaan. Seuraavassa on listattu kummankin vaihtoehdon hyviä ja huonoja puolia.

Taulukoton taitto

- Tiedostokoot pienenevät kun toistuvat rakenne ja tyylimäärityt on yhdellä tai (pienellä) joukolla erillisiä tiedostoja.
- Vie vähemmän tilaa kovalevyiltä ja tiedonsiirtokanavalta.
- Sivuston päivittäminen on helpompaa
- Ei ole enää tarpeen määrittää *valign top, border 0* ja *cellspacing 0* ja muita taulukoilla oletusarvoisesti toimivia määrittelyjä.
- Mahdollistaa ulkoasun pitämisen erillään rakenteesta
- Parantaa saatavuutta. Käyttäjä voi määritellä isomman kirjaisinkoon tai enemmän kontrastia sivuille.
- Mahdollistaa sivuston optimoinnin eri alustoille, kuten verkkojulkaisu, pda ja tulostus
- Huonona puolena divit tarvitsevat paljon ohjelmallisia kiertoteitä toimiakseen useassa eri selaimessa tai selainversiossa (cross-browser -tuki).

Johtopäätökset

CSS voi olla verkkosivujen tulevaisuutta, mutta vie vuosia ennen kuin taulukkopohjainen rakenne on täysin turha.

Taulukkotaitto

- Toimii hyvin, kun sivun koko on pieni, eikä ole tarvetta rakenteen dynaamisuuteen.
- Laajempien sivustokokonaisuuksien myötä HTML tiedostojen koko kasvaa
- Vaatii usein useita sisäkkäisiä taulukkorakenteita.
- Taulukoiden ylläpito vaatii enemmän resursseja
- Taulukkorakenteet on suunniteltu taulukoiden esittämiseen, ei kokonaisen sivun taittoon.
- Esteettömyysongelmat



KUVA 7. Kuva artikkelista: “Why tables for layout is stupid” (Seybold seminars)

2.4.6 Eri selainten huomioon ottaminen

Kun tehdään uusia verkkosivuja, on lähes välttämätöntä testata ne käyttämällä IE7:llä, koska suurin osa käyttäjistä selaa IE:llä (Korpela 2008, 56). Myös suuri osa virheistä CSS:n käytön yhteydessä on peräisin IE:stä.

”On jopa sanottu, että jos sivu näyttää toimivan IE:llä, mutta ei Operalla ja Mozillalla, niin kyseessä on yleensä IE:n virhe.” (Korpela 2008, 57.)

Mozilla Firefox:in ja muiden selainten yleistyessä, on kasvanut tarve ottaa huomioon eri selaimet ja selainversiot sivujen suunnittelussa ja testauksessa. Kun tarkoitetaan kaikkia selaimia tukevaa verkkosivua (tai myös verkkosovellusta), puhutaan tällöin termistä *cross-browser*. Vastaavasti, jos tarkoitetaan verkkosivua, joka tukee suurta osaa selaimia, termi on tällöin *multi-browser*. (Cross-browser – Wikipedia 2010.)

Miksi yhden selaimen tukema CSS-koodi ei kannata?

CSS on määrittelynsä mukaan pelkkiä *ehdotelmia* sivun ulkoasua varten, jotka voidaan jättää huomioitta kokonaan tai osittain. Jukka K. Korpela listaa kirjassaan (Korpela 2008, 58) syitä, miksi CSS-muotoiluja ei aina huomioida:

- kaikki selaimet eivät tue CSS:ää
- selaimet tukevat CSS:ää eri laajuisesti
- selaimen CSS-tuki on saatettu ottaa pois käytöstä
- käyttäjän tyyliohje voi ohittaa sivun tyyliohjeen
- CSS-määrittelyt eivät ole olleet vaikaita
- CSS-määrittelyt sallivat erilaisia selainkohtaisia tulkintoja
- selainten CSS-tuessa on virheitä.

Tämän vuoksi on hyvä ottaa huomioon ainakin yleisimmät selaimet ja selainversiot jo sivustoa laadittaessa ja erityisesti sivuston testaamisen yhteydessä. Jotkin nk. sudenkuopat (mm. tapaukset, joissa käyttäjän tyyliohje ohittaa sivun tyyliohjeet) voidaan välttää huolellisella CSS:n käytöllä. Tämän lisäksi on olemassa joukko ohjelmallisia kiertoteitä (*CSS hacks*), joissa hyödynnetään eri selainten erikoisominaisuuksia ja virheellisiä toimintoja. Jean-Baptiste Jung listaa blogissaan kymmenen eri käytäntöä tai CSS kiertotietä, joilla voidaan vaikuttaa selainyhteensopivuuteen (Jung 2009). Lisäksi sivulla on listattu joukko *cross-browser* -testaukseen soveltuvia testiohjelmia. Näistä testiohjelmista lisää seuraavassa luvussa. Lopuksi mainittakoon verkkosivu <http://www.quirksmode.org/>, mikä on erikoistunut selainyhteensopivuuden parantamiseen.

2.4.7 Työkalut sivuston teon apuna

Yksinkertaisimmillaan HTML- ja CSS-dokumentit voidaan kirjoittaa käyttämällä perustekstinkäsittelyohjelmaa, kuten esimerkiksi Muistiota (notepad). Kuitenkin on olemassa joukko työkaluja, ohjelmia joilla voidaan helpottaa kirjoitustyötä. Usein tärkein syy apuohjelmien käyttöön on rutiinityön välttäminen. Tämä korostuu kun kyseessä on iso sivusto, joka sisältää useita lähes samanlaisia sivuja. (Korpela & Linjama 2006.)

Sivun rakentamista helpottavat erilaiset generaattorit ja HTML-editorit.

HTML-editorilla tarkoitetaan ohjelmaa, joka osaa tuottaa HTML-koodia ja sen avulla voidaan suorittaa automaattisesti kotisivujen tekemisessä tarvittavia toimintoja, kuten kuvan liittäminen sivulle. (Miettinen 2001, 17.) Muita editoreissa esiintyviä ominaisuuksia on mm. ennakoiva tekstinsyöttö, eri värien käyttö eri elementeissä oikoluennan helpottamiseksi ja automaattinen merkinnän tarkistus. Tällaisia ohjelmia löytyy sekä ilmaisohjelmina (Crimsom) että maksullisina.

Generaattoriohjelmat ovat HTML-editoreja, joissa on laajemmat ominaisuudet ja, joilla voidaan generoida eli tuottaa kokonaisia sivuja tai sivujen osia graafisesti hiirellä valiten *wysiwyg* -tilassa. Tunnetuimpia tällaisia ohjelmia ovat maksulliset Microsoft FrontPage ja Macromedia Dreamweaver. Myös ilmaisohjelmia löytyy, mm. FrontPage Express, joka on ladattavissa Microsoft:n sivuilta. (Miettinen 2001, 17.)

Generaattoriohjelmien yksi heikkous on niiden tuottaman koodin vaikeaselkoisuus. Koodi on usein laajaa ja sisältää myös turhiakin määreitä. Tällöin sitä on hankala muokata ilman ko. ohjelmaa ja sivujen ylläpito kärsii.

Riippuen sivuston käyttökohteesta ja tarpeista on sivuston CSS- tyyliohjeet ja HTML-rakenteet hyvä testata ja tarkistaa. Sivuston toteuttajan on melkeinpä mahdotonta testata sivuaan lukuisilla eri selaimilla tai selainversiolla. Onneksi nykyään testaukseen ja tarkistamiseen on olemassa hyviä työkaluja. Näitä ovat eri selaimia ja selainversioita tukevat testiympäristöt (esim. BrowserShots, <http://browsershots.org/>) ja W3.org:n tarjoamat merkinnän tarkistusohjelmat eli validaattorityökalut (CSS:n merkinnän tarkistukseen oleva validaattori löytyy osoitteesta: <http://jigsaw.w3.org/css-validator/> ja HTML-merkinnän tarkistukseen tarkoitettu validaattori osoitteesta: <http://validator.w3.org/>).

Näiden generaattori-, editori- ja tarkistusohjelmien lisäksi on olemassa joukko selaimiin liitettäviä apuohjelmia (ns. *plug-in* tai *add-on*). Kaksi mainitsemisen arvoista apuohjelmaa ovat: Firebug ja Web Developer.

Firebug on Firefox selaimen saatava lisäosa. Firebug:lla on mahdollista editoida, valvoa ja testata niin CSS, HTML kuin Javascript- koodia. Firebug toimii selaimen yhteydessä, ja ilmoittaa mm. kun kohtaa sivulla virheen Javascriptissä. Firebug:lla on helppo etsiä virhe ja korjata se kädenkäänteessä.(Firebug 2010.)
Firebug on ladattavissa osoitteessa: <http://getfirebug.com/> .

Web Developer on niin ikään Firefox:n lisäosa. Web Developer tarjoaa laajan valikoiman työkaluja verkkosivujen ja –sovellusten tekemiseen. Web Developer luo selaimen uuden työkaluvalikon, jossa on valikot mm. CSS:n, evästeiden (*cookie*), lomakkeiden ja kuvien hallintaan.(Web Developer 2010.)
Web Developer on ladattavissa osoitteessa: <http://chrispederick.com/work/web-developer/> .

3 CASE STUDY – CUSTOMIZATION OF INDICO’S CONFERENCE WEB PORTAL

Tässä pääluvussa keskitytään insinööri työn käytännön osuuteen. Aluksi esitellään vanha Indicon versio, jota projektin myötä paranneltiin. Tämän jälkeen perehdytään ongelmiin, joihin etsittiin ratkaisua ja kerrotaan miten ongelmaa ryhdyttiin ratkaisemaan. Lopuksi selostetaan toteutettu ratkaisu ja tehty työ vaiheittain. Kaikki työssä esitetyt funktiot on kirjoitettu Python –skriptikieltä käyttäen. Pythonia oli käytettävä, koska koko Indico-järjestelmä on luotu sillä. Kuitenkin ratkaisuna esitelty toteutus on myös mahdollista kirjoittaa muillakin skriptikielillä.

3.1 Indico versio 96.2

Indicon versio 96.2 otettiin käyttöön 27 kesäkuuta 2008. Versio toi mukanaan useita uusia toiminnallisuuksia, mutta jätti sivujen rakenteen vanhanmalliseksi. Seuraavissa luvuissa käydään läpi version 96.2 konferenssiosion HTML-rakenne, sivun asemointi ja muokkaukset, joita oli mahdollisuus tehdä.

3.1.1 Konferenssi osion sivuston vanha rakenne

Konferenssisivujen HTML-rakenne oli toteutettu taulukkotaittoa käyttäen. Tämä tarkoitti sitä, että HTML-rakenne koostui sisäkkäisistä taulukoista ja oli näin ollen suurikokoinen, vaikeaselkoinen ja vaikea ylläpitää. Lisäksi kiinteä taulukkotaitto mahdollistaa vain yhden taittorakenteen sivuille. Jos sivuston luoja haluaa käyttää taittorakennetta, joka tuottaa menu-valikon vaakatasoon heti otsakeosion alle, ei se ole vanhan HTML-rakenteen avulla mahdollista.

3.1.2 Konferenssi osion sivuston vanha asemointi

Vanhassa versiossa Indico tarjosi vain yhden asemoinnin mallin konferenssisivuille. Sivut koostuivat neljästä (4) pääalueesta. Ylävasemmalla oli tila logolle. Yläoikealla oli otsake-alue, joka oli jaettu konferenssin nimelle, tapahtumapaikalle ja -ajalle. Sivujen alaosa oli jaettu siten, että vasemmalla puolella oli kiinteä linkkilista ja oikealle puolelle oli varattu alue varsinaiselle konferenssia koskevalle informaatiolle. (Kuva 8)



KUVA 8. Konferenssisivujen asemointi

Logon puuttuessa paikalla on vain tyhjä valkoinen osio (kuva 8). Otsakealueen koko skaalautui otsikon koon eli pituuden mukaan. Kuitenkaan otsakealue ei skaalautunut logon koon mukaan, vaan jos logo oli isompi kuin sille varattu alue, näkyi se koko koossaan muokaten sivujen rakenteen epäsuhtaiseksi (kuva 9). Kiinteä linkkilista kasvoi linkkien määrän mukaan.



KUVA 9. Liian suuren logon vaikutus asemoinnissa

3.1.3 Konferenssi osion sivuston vanha muokkaus

Jo alusta alkaen Indicoa on markkinoitu sen personoitavilla tapahtumasivuilla (Baron 2005). Kuitenkin vielä versiossa 96.2 tämä personointi on hyvin rajoittunutta.

Versio 96.2 tarjoaa käyttäjälle kiinteän rakenteen lisäksi vain kolme eri mahdollisuutta ulkoasun muutoksiin. Nämä ovat:

- Oman logon käyttö logolle varatussa paikassa, sivun vasemmassa yläkulmassa.
- Otsikossa käytetyn fontin värin vaihto.
- Otsikon taustavärin muuttaminen. (Kuva 10, Kuva 11)

The screenshot shows a web interface titled "Customization of conference display". It has a sidebar on the left with links: "Title background color", "Title text color", and "Display Search Box". The main area has a heading "Select an option in order to customize it" and a section titled "Logo". Under "Logo", there is a "Preview:" label, a text input field, a "Selaa..." button, and a "save the logo" button.

KUVA 10. Mahdollisten muutosten asetukset hallinnointisivuilla

The screenshot shows a website for the "CERN workshop on Innovations in Scholarly Communication (OAI6)". The header is orange with the title in white. Below the title, it says "17-19 June 2009" and "University of Geneva". There is a search bar with a "Search" button. On the left, there is a navigation menu with links: Home, Announcements, Program, Practical Information, Accommodation & Transport, Contribution List, Registration Form, Speaker index, Social events, Organisation, Sponsors, and Previous OAI Workshops. The main content area has a "Home" heading and text about the workshop, including its format, aims, and previous editions.

KUVA 11. Mahdolliset muutokset sivun ulkoasuun

3.1.4 Ongelmat vanhassa versiossa

Tapahtumaportaalin nykyinen versio on vanhentunut, eikä enää vastaa käyttäjien tarpeisiin. Ongelmia tuottaa erityisesti portaalin rakenteen ja tyylin muokattavuus tai varsinkin muokattavuuden vajavaisuus. Nykyään yhä useammalla tapahtumalla ja instituutiolla on omat kotisivut ja etukäteen suunniteltu sivuston imago. Tähän liittyy käytettävät rakenne, kirjasinlajit sekä värikartta. Indicon konferenssiosion sivuja ei voida muokata ulkoasultaan tai rakenteeltaan näiden nykyisten vaatimusten mukaiseksi.

Muutosmahdollisuuksien vähyys on johtanut siihen, että nykyään monista tapahtumista on olemassa kaksi sivustoa;

- Toinen käytännön funktioihin, kuten materiaalin etukäteisarviointi ja materiaalin tiivistelmien lähettäminen (Indico)(liite 2)
- Toinen tapahtumien ulkoasun esittämiseen (Tapahtuman/instituution omat kotisivut)(liite 2)

Näin ollen sivustoilla vieraileva joutuu pahimmassa tapauksessa siirtymään sivustolta toiselle tarpeidensa mukaan.

Toinen nykyisen version hankaluuksista on se, että nykyiset sivut eivät skaalaudu käytettävän selainikkunan koon mukaan. Tämä johtaa siihen, että käytettäessä pienempää selainikkunaa tai näyttölaitetta, vain osa sivustosta on näkyvässä. Tämä haittaa sivuston selattavuutta.

3.2 Ratkaisuvaihtoehdot

Tässä luvussa käydään läpi, miten ongelmakohtia aloitettiin ratkaisemaan. Aluksi kerrotaan taustatyöstä ja tutkimuksesta, joilla kartoitettiin tapahtumien ja instituutioiden sivuille asettamia ulkoasullisia vaatimuksia. Tämän jälkeen selostetaan, millaisia eri ratkaisumalleja löydettiin ja miten päädyttiin lopulliseen ratkaisumalliin.

3.2.1 Taustatyö ja tutkimus

Jotta voitiin aloittaa sivuston ongelmien ratkaiseminen, täytyi aluksi tehdä taustatutkimusta. Tutkimus aloitettiin vertailemalla nykyisiä konferenssien ja tapahtumien sivustoja, niin Indicon sisällä kuin ulkopuolella, jotta saatiin selville millaisia ominaisuuksia uudelta versiolt vaadittaisiin. Vertailukohtina käytettiin niin sivuston rakenteeseen, yleiseen ulkoasuun kuin myös sisältöön liittyviä ominaisuuksia.

Vertailukohdiksi valittiin:

- sivuston rakenne
- sivuston menutarjottimen rakenne
- otsikkokentän rakenne
- sivustolla käytettyjen kuvien lukumäärä
- sivuston väriskaala.

Ennenkaikkea tutkimuksella pyrittiin selvittämään se, kuinka monta ja millaisia muutoksia tarvitaan Indicon tarjoamiin perussivuihin, jotta saadaan aikaan samankaltaiset sivut. Tämän lisäksi tutkittiin, millaisia mahdollisuuksia muutoksiin Indicon nykyinen versio tarjoaa.

3.2.2 Erilaisia ratkaisumalleja

Ratkaisun pohjana toimi taustatyönä tuotetun tutkimuksen tulokset sekä työtiimin omat toiveet. Pääkohtana oli luoda ratkaisumalli, jolla on mahdollista toteuttaa tarvittavat muutokset Indicon tarjoamien sivujen rakenteeseen ja ulkoasuun. Lisäksi oli toiveina *wysiwyg*-editori, jossa käyttäjä voisi asemoida sivuston rakenteen osia *drag-and-drop*-ominaisuutta hyödyntäen. Projektissa lähdettiin hakemaan MVC-mallin tapaista ratkaisua. Niin että sivuston ulkoasua ja rakennetta voidaan vaihtaa muuttamatta Indicon koneistoa.

Vaihtoehtoisiksi ratkaisuiksi ehdotettiin:

- Pythonilla tuotettu editori, jolla muodostetaan sivut
- Lisäämällä nykyiseen versioon vain muutamia uusia ominaisuuksia
- Javascripteillä tuotettu rakenne
- CSS/div-rakenteella tuotettu rakenne

3.2.3 Toteutettu ratkaisu

Ajatus erillisestä editorista jossa on *wysiwyg*- ja/tai *drag-and-drop*-ominaisuus olisi ollut mainio ratkaisumalli. Koska projekti oli toteutettava tiukassa aikataulussa, päädyttiin käyttämään niitä tekniikoita ja kokemuksia, joita kirjoittajalla oli jo entuudestaan. Näin ollen päädyttiin käyttämään CSS-tyylisivustoja rakenteen ja ulkoasun muokkaukseen. Kuitenkin uutta versiota toteutettaessa pidettiin avoinna mahdollisuus myöhempään päivitykseen, jolla voidaan mahdollistaa myös sivuston muokattavuus *drag-and-drop*-ominaisuutta käyttämällä.

3.3 Tehdyt muokkaukset uuteen versioon

Seuraava luku käsittelee kaikkia niitä muokkauksia ja ominaisuuksien lisäyksiä, joita tuotettiin projektin myötä uuteen versioon. Aluksi käydään läpi sivujen HTML-rakenteeseen tehdyt muutokset. Tämän jälkeen käsittelyssä on hallinnointisivut ja niihin tehdyt uudistukset ja muutokset. Lopuksi käydään läpi luodut CSS tyyliohjeet ja toiminnallisuudet mahdollistavat funktiot.

3.3.1 Tehdyt muokkaukset konferenssisivujen HTML-rakenteeseen

Koska vanha HTML-rakenne oli toteutettu käyttäen taulukkotaittoa, oli ensimmäisenä työvaiheena vanhan HTML-rakenteen korvaaminen käyttämällä *div*-elementtejä. Taulukkotaittoakin olisi ollut toimiva mahdollisuus joissakin sivuston tarpeissa, mutta koska haluttiin ratkaisuun, jossa käyttäjä voi muokata sivuston rakennetta ja ulkoasua ilman HTML-koodaamista, tarvittiin taulukotonta rakennetta. *Div*:it matkivat taulukoilla luotua rakennetta, kuitenkin niin, että sisäkkäisten elementtien määrä väheni radikaalisesti. (liite 1).

Uusi HTML-rakenne koostuu ylätasen elementistä nimeltä conf, jonka alla ovat kaikki sivuston pääelementit. Conf-elementin alue on koko sivun alue ja se on jaettu kahteen suureen säiliöön: otsikko-osioon (header) ja päärakenteeseen (main content).

Header sisältää tilat logolle, otsikolle sekä paikka- ja aikatiedoille. Koska nyt kaikki otsikko-elementit ovat tämän header-säiliön sisällä, voidaan logo taikka taustakuva laajentaa käsittämään koko otsikkoaluetta. Esimerkiksi voidaan luoda taustakuva, jossa tapahtuman logo vaihtuu liukuen otsikon pohjaväriksi. Nyt on myös mahdollista skaalata logon koko suhteessa otsikkoon .

Päärakenne (main content) on jaettu kahteen pääalueeseen, joista ensimmäinen on menu-alue. Menu on yksi monimutkaisemmista rakenteista koko sivulla. Mutta näin ollen on tarjoutunut mahdollisuus niin menun rakenteen kuin paikan muokkaamiseen. Menu tarjoaa eri tyylikokonaisuudet valituille valikko-elementeille, tavallisille valikko-elementeille ja alavalikko-elementeille. Koska alavalikkoelementit ovat uuden version myötä dynaamisia, antaa se tilaa itse sisällölle, esimerkiksi aikatauluille. Toinen päärakenteen osa on varattu sisällölle itselleen. Sisältönä voi olla aikataulu, tapahtuman keskeiset tiedot, lähetetty materiaali tai vaikka ilmoittautumislomake.

3.3.2 Tehdyt muokkaukset hallinnointisivuille

Hallinnointisivuissa (Management area) on nyt kaksi uutta osiota. Osioden nimet ovat CSS ja Picture Uploader.

CSS osio sisältää (Kuva 12):

- Minilomakkeen CSS-tiedoston lataamiseen Indicoon
- Esikatseluikkunan, josta voi selailla ladattua CSS-tiedostoa
- Poistamisfunktion, jolla ladatun CSS-tiedoston saa pois käytöstä
- Esikatselukuvat valmiina olevista rakenne-pohjista. Jokainen kuva toimii myös linkkinä. Kuvaa klikkaamalla käyttäjä pääsee esikatselu-sivustolle, joka käyttää kyseessä olevaa rakennetta. Esikatseluosiota kerrotaan lisää luvussa 3.3.5.



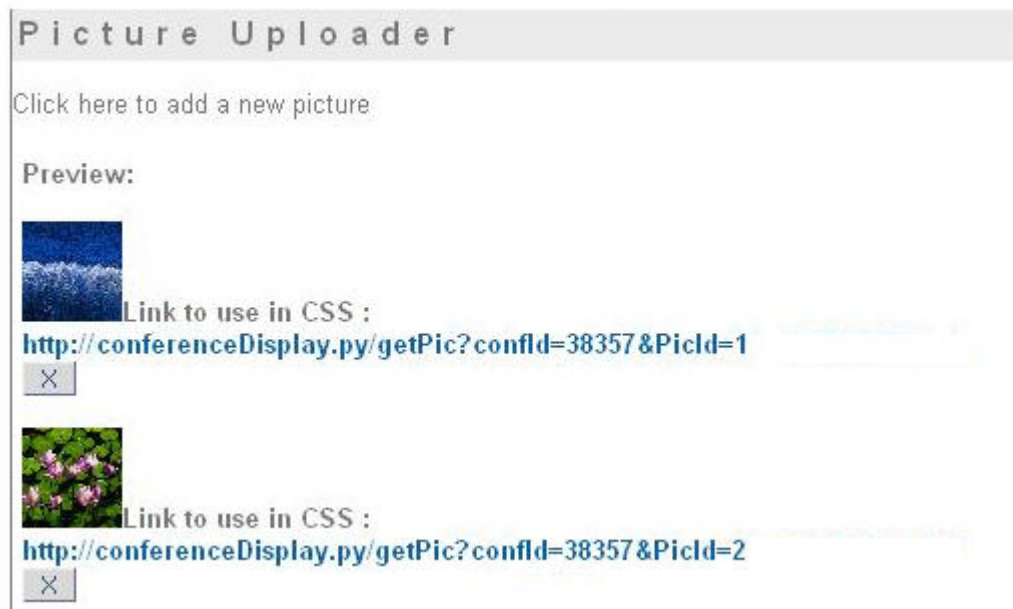
KUVA 12. CSS- oso hallinnointisivuilla

Kuvien latausosio (Picture Uploader) sisältää:

- Minilomakkeen kuvien lataamista varten. (Kuva 13)
- Kuvaelementtejä, jotka sisältävät (kuva 14):
 - Esikatselukuvakkeen kuvasta.
 - Kuvan yksilöllisen URL:n jota käyttämällä käyttäjä voi viitata kuvaan CSS-tiedostossa.
 - URL toimii myös linkkinä, josta voi nähdä kuvan oikeassa mittakaavassa.
 - Painike kuvan poistamiseen.



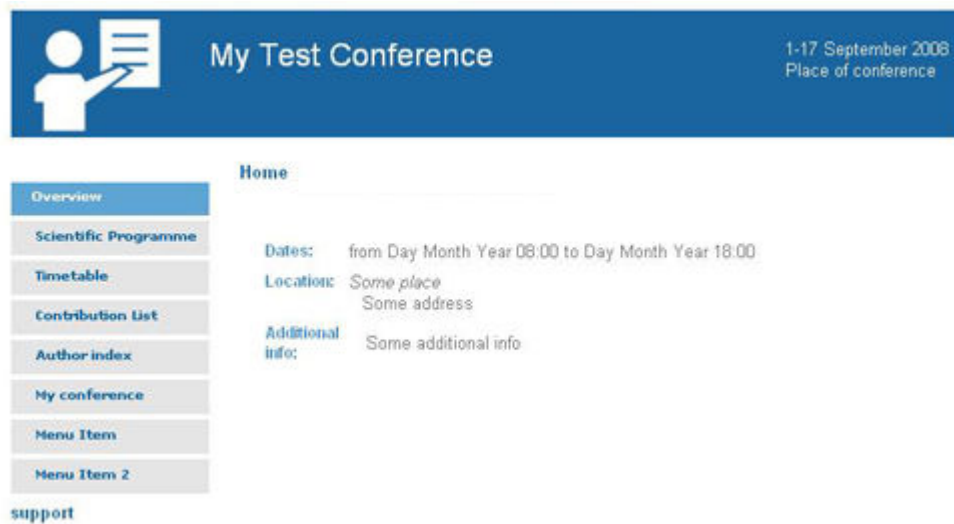
KUVA 13. Kuvien latausosio hallinnointisivuilla



KUVA 14. Kuvien latausosion luomat kuvaelementit

3.3.3 Oletusarvoisen rakenteen ja ulkoasun CSS-tiedosto

Sivuston oletusarvoista rakennetta ja ulkoasua varten luotiin CSS-tiedosto nimeltä Conf_Basic, mikä luo konferenssisivuston perusrakenteen ja ulkoasun (Kuva 15). Tämä rakenne-ulkoasu- yhdistelmä on aina oletusarvoisesti käytössä, kun luodaan uusi tapahtuma. Sitä käytetään, ellei sivun luoja lataa omaa muokattua CSS-tiedostoa tai valitse erillaista rakennepohjaa. Conf_Basic määrää elementtien asemoinnin sivustolla sekä oletusarvoiset värit eri osioille ja kirjaisimille.




KUVA 15. Havainnekuva oletusarvoisesta ulkoasusta

3.3.4 Esivalmistetut rakennepohjat

Projektin päättyessä Indico tarjoaa kolme erilaista esivalmistettua pohjaa sivuston rakenteelle, yksi kutakin sivuston erilaista rakennetyyppiä varten. Rakennetyypit ovat:

- Pystysuora valikko sivuston vasemmassa reunassa.
- Pystysuora valikko sivuston oikeassa reunassa.
- Vaakataso valikko otsikkoalueen alla. (Kuvat 16a-c)

CSS-pohjat voidaan ottaa suoraan käyttöön tai ne voidaan ladata ja tehdä tarvittavat muokkaukset. Muokkauksia varten CSS-tiedostoissa on omat kohdat, jotka sivuston luoja voi täyttää tai muokata haluamakseen. CSS-tiedostot ovat myös hyvin kommentoituja ja sisältävät yksityiskohtaiset opastukset ja esimerkit muokkausten tekemiseen.



My Test Conference

1-17 September 2008
Place of conference

Overview

Scientific Programme

Timetable

Contribution List

Author index

My conference

Menu Item

Menu Item 2

Home

Here you can test different layouts.

- In the upper left corner is a selection box. Choose the template and see the results
- Next to the selection box is link, where you can upload the correspond CSS file. Just click the right mouse button and choose 'save link as'.
- You can also take the layout in use right away just clicking the 'take template in use' - button.
- You can go back to management area, just using the link 'Back to management area'.

Dates:

from Day Month Year 08:00 to Day Month Year 18:00

Location:


Some place
Some address

Additional info:

Some additional info

support

KUVA 16a. Menun sijoitus vasempaan reunaan



My Test Conference

1-17 September 2008
Place of conference

Home

Here you can test different layouts.

- In the upper left corner is a selection box. Choose the template and see the results
- Next to the selection box is link, where you can upload the correspond CSS file. Just click the right mouse button and choose 'save link as'.
- You can also take the layout in use right away just clicking the 'take template in use' - button.
- You can go back to management area, just using the link 'Back to management area'.

Dates:

from Day Month Year 08:00 to Day Month Year 18:00

Location:

Some place
Some address

Additional info:

Some additional info

Overview

Scientific Programme

Timetable

Contribution List

Author index


My conference

Menu Item

Menu Item 2

support

KUVA 16b. Menun sijoitus oikeaan reunaan



My Test Conference

1-17 September 2008
Place of conference

Overview

Scientific Programme

Timetable

Contribution List

Author index

My conference

Menu Item

Menu Item 2

Home

Here you can test different layouts.

- In the upper left corner is a selection box. Choose the template and see the results
- Next to the selection box is link, where you can upload the correspond CSS file. Just click the right mouse button and choose 'save link as'.
- You can also take the layout in use right away just clicking the 'take template in use' - button.
- You can go back to management area, just using the link 'Back to management area'.

Dates:

from Day Month Year 08:00 to Day Month Year 18:00

Location:

Some place
Some address

Additional info:

Some additional info

support

KUVA 16c. Menun sijoitus suoraan otsakealueen alle

3.3.5 Funktiot CSS:n käyttöönottoa varten

CSS-tiedostojen käyttöönottoa varten luotiin joukko erilaisia funktioita, joista osa sijaitsee suoraan hallinnointisivuilla ja osa esikatselu-osiossa. Hallinnointisivuilla sijaitsee funktiot CSS:n lataamiseen Indicoon. Lataaja ottaa tapahtuman yksilöivän tunnusteen (confid) parametrinä ja tallentaa CSS-tiedoston tietokantaan käyttämällä tätä tunnustetta identifikaationa. Konferenssisivujen otsikkokentän luonnin yhteydessä on funktio, joka hakee ladatun ja tallennetun CSS-tiedoston käyttöön sillä ehdolla, että ko. confid:ä vastaava CSS-tiedosto on ladattu tietokantaan. Näin ollen vain kyseessä olevan tapahtuman manageri voi tehdä muutoksia ko. tapahtuman CSS-tiedostoon. CSS-tiedoston esikatselua varten luotiin funktio, joka luo yksilöllisen URL:n ladattuun ja tallennettuun CSS-tiedostoon, siten että sitä voidaan esikatsella ja selaila esikatseluikkunassa.

3.3.6 Esikatselusivu (Preview page)

Esikatseluosio (preview page) on alue, missä tapahtumasivujen luoja voi vertailla eri rakenteiden välillä. Esikatseluosio toimii hiekkalaatikko- periaatteella. Esikatseluosion sivu on täydellinen kopio varsinaisesta tapahtumasivusta, mutta kuitenkin niin, ettei siihen tehdyt muutokset vaikuta itse varsinaiseen tapahtumasivustoon. Esikatseluosiossa käyttäjä voi selaila eri CSS-tiedostojen aikaansaannoksia, tämän jälkeen valita mieluisin, ottaa se heti käyttöön tai ladata koneelle muutosten tekemistä varten. Esikatseluosio on rakennettu täysin erilleen aitojen sivustojen rakennuksesta, siten että sitä voidaan myöhemmin käyttää myös muuhun testaamiseen.

3.3.7 Esikatselusivua varten luodut funktiot

Esikatselusivu sisältää työkalupalkin ja sen alla olevan työtilan, joka näyttää aivan tavalliselta konferenssisivuilta sisältäen samat peruselementit. Työkalupalkissa on valikko, josta käyttäjä voi valita haluamansa pohjan. Kun käyttäjä tekee muutoksen valikon arvoon (vaihtaa valmispohjaa), preview page rakentaa sivuston käyttäen vastaavaa ulkoasua ja rakennetta. Myös kaikki työkalupalkin toiminnallisuudet päivittyvät koskemaan kyseessä olevaa pohjaa. Pohjan valinnan jälkeen käyttäjä voi ladata CSS-tiedoston koneelle luotua linkkiä käyttäen. Mikäli käyttäjä ei halua tehdä muutoksia CSS-tiedostoon, on käyttäjällä mahdollisuus ottaa kyseessä oleva pohja käyttöön työkalupalkissa sijaitsevaa nappia painamalla. Nappia painamalla käynnistyy funktio, joka tallentaa valitun pohjan tietokantaan, kuten ladatun CSS-tiedoston, käyttäen confld – identifikaatiota. Tallennuksen (käyttöönoton) jälkeen käyttäjä uudelleenohjataan hallinnointisivuille. Tämän jälkeen valittu pohja on käytössä samoin kuin ladattu CSS-tiedosto. Suoraa paluuta hallinnointisivuille varten on työkalupalkissa *palaa takaisin (back)*-linkki.

3.3.8 Kuvien latausta varten luodut toiminnallisuudet

Hallinnointisivuille on luotu Picture uploader- niminen osio, missä käyttäjä voi ladata ja tallentaa kuvia tietokantaan. Kun käyttäjä klikkaa *lisää kuva (add new picture)*-linkkiä, avautuu elementti jolla voi selata ja tallentaa kuvan tietokantaan.

Tallennuselementti kutsuu next_pic_id – nimistä funktiota, joka tarkistaa seuraavan vapaan paikan kuvalle. Tämä vapaa paikka voi sijaita myös olemassa olevien kuvien välissä, mikäli siltä paikalta on poistettu kuva. Jos vapaata paikkaa ei löydy kuvien välistä, luo next pic id- funktio uuden aihion viimeisen olemassa olevan jälkeen.

Picture uploader tallentaa kuvat tietokantaan käyttäen apunaan picId ja confid -tunnisteita. Kuvia voidaan nyt tallentaa useampia entisen yhden logokuvan sijaan.

Kun käyttäjä on tallentanut kuvia, on jokaista kuvaa varten luotu yksilöllinen elementti hallinnointisivuille. Enemmän elementtien rakenteesta kerrotaan luvussa 3.3.2. (Kuva 14). Elementit luodaan käyttämällä funktiota, joka tarkastaa tietokantaan tallennettujen kuvien lukumäärän ja luo jokaista kuvaa vastaavan elementin. Jokainen näistä kuvaelementeistä sisältää poisto-nappulan, jolla kuva poistetaan tietokannasta ja alkion arvoksi asetetaan *None*.

Kuvien esittämistä ja käyttöä varten on luotu funktio nimeltä `getPic`. Tämä funktio käyttää `picId` ja `confId` – tunnisteita hakiessaan ja esittäessään kyseessäolevan kuvan. Funktio luo kuvaa vastaavan yksilöllisen URL:n, johon käyttäjä voi viitata CSS-tiedostossa ja näin ottaa kuvan käyttöön.

3.4 Testaus

Kaikki luodut funktiot ja rakenteet testattiin Indicodev-testausympäristössä, mikä vastaa kaikilta ominaisuuksiltaan varsinaista käyttöympäristöä. Tämän lisäksi kaikki sivustoon liittyvät elementit (rakenne ja dynamiikka) testattiin käyttämällä useimpia eri selainmalleja ja –versioita. Myös sivuston toiminta eri resoluutioita käyttämällä testattiin.

Testauksessa käytetyt selaimet ja selainversiot:

- Internet Explorer 7
- Internet Explorer 8 beta
- Mozilla Firefox 2.0.0.16
- Mozilla Firefox 3.0.1
- Opera 9.52
- Safari 3.1.2

Testaus tapahtui matkimalla käyttäjän työjärjestystä sivustoa luotaessa.

Testaus sisälsi:

- Uusien konferenssisivujen luominen
- Valmiin pohjan valinta
- Valmiin pohjan käyttöönottaminen
- Käytössä olevan CSS-tiedoston poisto
- CSS-tiedoston lataaminen, muokkausten tekeminen ja käyttöönotto
- Kuvien tallennus
- Kuvien käyttöönotto ja poisto.
- CSS-tiedoston muokkaus käyttäjälle annettujen ohjeitten mukaisesti

Nämä testaukset toteutettiin kaikille kolmelle olemassa olevalle rakenteelle. Ulkoasut testattiin ja tarkistettiin aiemmin mainitun listan selaimilla. Samoin tehtiin myös toiminnallisuuksien kanssa.

Testauksessa ei löytynyt yhtään vakavaa virhetilannetta. Kaikki toiminnallisuudet toimivat kuten niiden pitikin. Testauksessa ilmeni joitakin virheitä sivun elementtien sijoittelussa, jotka johtuivat lähinnä eri selainten yhteensopimattomuuksista. Nämä virheet saatiin korjattua ja uudelleen testattua ennen projektin loppua. Kehitystyö ja testaus sivuston rakennuksen osalta jatkuu vielä projektin jälkeen.

3.5 Turvallisuus

Tietoturvallisuus on aina tärkeä asia pitää mielessä. Tämä otettiin huomioon myös tätä projektia suunniteltaessa ja toteutettaessa. Seuraavassa on kerrottu tietoturvan takaavia seikkoja.

Ensinnäkin käyttäjän on kirjauduttava palveluun ennen kuin hän voi tehdä mitään muokkauksia sivustoon. Tämä myös varmistaa sen, ettei kukaan muu kuin sivuston luoja tai manageri pysty tekemään muutoksia sivustoon. Kirjautuneella käyttäjällä on ne oikeudet sivuston muokkaukseen kuin sivuston luoja on ne asettanut. Kaikki käyttäjän muokkaamat osa-alueet kuten CSS-tiedosto ja ladatut kuvat identifioidaan kyseistä konferenssia käsittäviksi. Näin ollen niihin tehdyt muutokset vaikuttavat vain kyseessä olevaan konferenssiin. CSS-tiedoston käyttö ei muokkaa HTML-rakennetta millään tavoin ja kaikki muokkaukset CSS-tiedostoa käyttäen on mahdollisia poistaa käytöstä. Koska CSS-tiedosto otetaan käyttöön käyttämällä yksilöllistä URL:ia ja `<link rel="stylesheet" type="text/CSS">` -elementtiä, ei muun materiaalin lataaminen ja käyttöönotto vaikuta sivuston toimintaan. Ja jälleen kerran, muutokset vaikuttavat vain kyseessä olevaan konferenssisivustoon, eikä siten vaikuta koko Indico-järjestelmää.

4 JOHTOPÄÄTÖKSET

Tässä luvussa esitellään projektin myötä saavutetut lopputulokset. Lisäksi pohditaan, mitä tehdyn projektin lopputulokset mahdollistavat tulevaisuudessa, varsinkin verkkosovelluksen jatkokehityksen kannalta. Luvussa esitellään myös keväällä 2010 käyttöönotettu uusi Indico- versio ja siinä mukana olleet tämän projektin tuotokset. Aivan luvun lopussa on tekijän omat loppusanat.

4.1 Projektin lopputulokset

Tekemäni projektin tuloksena:

- Konferenssisivuilla on nyt kolme eri rakennevaihtoehtoa.
- Sivustot ovat nyt skaalautuvia. Muutokset selainikkunan kokoon eivät vaikuta sisällön näkymiseen, tietyin rajoituksin.
- Käyttäjä voi muokata sivuston rakenteellisia osia, saaden siten paremmin personoidun ulkoasun sivustolle.
- Käyttäjä voi ladata useampia kuvia ja ottaa ne käyttöön sivustolla.
- Koska HTML-rakenne on nyt yksinkertaisempi, on sivuston jatkokehitys ja ylläpitotyö helpompaa.

4.2 Indico- projektin tulevaisuus

Työ Indico -projektin parissa jatkuu yhä muiden osalta. Tulevaisuuden kannalta tekemäni sivuston uusi toteutus tarjoaa jatkokehitykselle useita vaihtoehtoja. Yksi on päivittää käyttöön *drag-and-drop*-sivueditori. Tämän lisäksi on mahdollisuus luoda iso valikoima erilaisia, valmiiksi toteutettuja ulkoasuja ja rakenteita tapahtumasivun laatijan valittavaksi. On myös mahdollista jatkokehittää *preview page*- esikatselutilaa laajemmaksi sivuston testausympäristöksi, ja näin helpottaa sivuston laatijan työtaakkaa.

4.3 Indico versio 0.97b2

Uusin versio Indicosta (versio 0.97b2) otettiin käyttöön 14.3.2010. Se on selailtavissa verkko-osoitteesta: <http://indico.cern.ch/> . Uuden version myötä tekemäni uudistukset ja muokkaukset otettiin käyttöön tietyin muutoksin:

- Oletusarvoinen ulkoasu on saanut uuden ilmeen. Tämä oli odotettavissa, sillä luomani ulkoasu oli lähinnä yhteisellä päätöksellä tehty luonnosversio, joka vain edeltää varsinaista ulkoasua.
- HTML-rakenne oli säilynyt muutoin samana, paitsi varsinaisen sisällön rakenne oli korvattu taulukkorakenteella. Tämä on varmasti hyvä ratkaisu, koska sisällön laajuudesta ei ole varmuutta ja näin ollen taulukkorakenne on varmempi säilyttämään muotonsa.
- Myös muita pienempiä muutoksia HTML-rakenteeseen oli tehty, joista ainakin yksi oli virheellinen (sisäelementin sisälle oli asetettu lohkotason elementti).
- Hallinnointisivuilla CSS-osion ja kuvien latausosion ulkoasut olivat kokeneet muutoksen. Muuten olivat säilyneet miltein muuttumattomina.
- Iloisena huomiona totesin, että tekemäni esikatseluosio oli otettu käyttöön. Vaikka kyseessä oli siis tekemäni lisäominaisuus, ei asiakkaan tilaama ominaisuus.
- Asiakkaiden eli tapahtumasivujen laatijoiden puolesta käyttöönotto on sujunut hyvin ja useat tapahtumat ovat saaneet oman persoonallisen ulkoasun.

Erikoismainintana mainittakoon ICHEP 2010 –tapahtuma, jonka sivu löytyy osoitteesta <http://indico.cern.ch/conferenceDisplay.py?confId=73513> . Kun tätä sivua vertaillaan tapahtuman omiin sivuihin (löytyy osoitteesta: <http://www.ichep2010.fr/>), todetaan, että ulkoasu on liki identtinen halutun ulkoasun kanssa. Joissakin tapauksissa (esim. punaiset pallokuvat menu-elementtien edessä) Indicon ulkoasu toimii alkuperäistä paremmin. ICHEP 2010 –tapahtuma todentaa, mitkä mahdollisuudet uudistettu tapahtumaosion käyttöliittymä tarjoaa. Ja luultavammin pian ei tarvita enää kahta erillistä sivustoa per tapahtuma. Tämän puolesta voin sanoa, että osuuteni Indico-projektin parissa oli menestys.

4.4 Loppusanat

Työskentely CERNissä oli upea ja ainutlaatuinen kokemus, jota en unohda. Asiaan ei vaikuta pelkästään mahdollisuus työskennellä yhdessä maailman suurimmista tieteellisistä instituutioista, vaan myös mahdollisuus työskennellä monikulttuurisessa ympäristössä jossa jokainen tekee työtä tieteen, tekniikan ja yhteisen hyvän puolesta.

Myös tehty työ oli paras mahdollinen projekti minulle. Olen aina halunnut oppia uutta ja työskentely verkkosovellusten parissa on ollut aina lähellä minua. Tässä projektissa opin uusia tekniikoita ja työkaluja sekä uusia tapoja työskennellä ja suunnitella. Projektissa oli tarpeeksi haasteita ollakseen jatkuvasti mielenkiintoinen ja opettava, mutta kuitenkin tarpeeksi helppo, että se oli toteutettavissa tieto-taito-tasollani sekä vaaditussa ajassa.

Syy siihen miksi tämä opinnäytetyö valmistui puolitoista vuotta projektin päättymisen jälkeen on se, että halusin odottaa uuden version käyttöönottoa, jotta voin todentaa tekemäni työn niin seminaarissa, kuin tässä kirjallisessa osuudessakin. Sitä odotellessani keskityin muihin projekteihin ja kartutin omaa tieto-taito-tasoani verkkosovellusten- ja sivustojen laadinnassa. Uskon, että odotusaika kypsensi minua tietotekniikan osaajana ja vaikutti positiivisesti myös tähän työn kirjalliseen osuuteen.

5 LÄHTEET

About Indico [verkkodokumentti] CERN

[viitattu: 24.4.2010] Saatavissa: <http://indico.cern.ch/about.py>

Baron, Thomas 2005. Indico: An Event Management Software (and more) – HEPiX Fall meeting 2005. [PowerPoint- kalvosarja]

Berners-Lee, T., Masinter, L., McCahill, M. 1994.

Uniform Resource Locators (URL) [verkkodokumentti] IETF

Julkaistu: Joulukuu 1994 [viitattu 23.4.2010]

Saatavissa: <http://www.ietf.org/rfc/rfc1738.txt>

CDS Indico Overview 2005 [verkkodokumentti] CERN

Päivitetty: 8.8.2005 [viitattu: 24.4.2010]

Saatavissa: <http://cdsware.cern.ch/indico/>

Cross-browser – Wikipedia, the free encyclopedia[verkkodokumentti]

Päivitetty: 16.4.2010 [viitattu: 28.4.2010]

Saatavissa: <http://en.wikipedia.org/wiki/Cross-browser>

Cascading Style Sheets, level 2

CSS2 Specification [verkkodokumentti] W3C

Päivitetty: 11.4.2008 [viitattu: 20.4.2010]

Saatavissa: <http://www.w3.org/TR/2008/REC-CSS2-20080411/>

Ferreira, Jose Pedro 2006/2007.

Improving the Indico Framework at the European Organization for Nuclear Research, Internship Report [PDF]

Firebug [verkkodokumentti]

[viitattu 28.4.2010]

Saatavissa: <http://getfirebug.com/>

Jung, Jean-Baptiste. 15+ techniques and tools for cross browser CSS coding

[verkkodokumentti] [viitattu: 28.4.2010]

Saatavissa: <http://www.catswhocode.com/blog/15-techniques-and-tools-for-cross-browser-css-coding>

Korpela, Jukka K. 2003. CSS-tyylit. Porvoo: WS Bookwell

Korpela, Jukka K., Linjama Tero 2003. Web-suunnittelu.

[e-kirja] Porvoo: WS Bookwell

Korpela, Jukka K. 2008. CSS verkkosivujen muotoilussa. Porvoo: WS Bookwell

Miettinen, Petri 2001. Kotisivujen tekeminen. [e-kirja] Docendo: Jyväskylä

Peltomäki, Juha 2006. Web-selainohjelmointi. Porvoo: WS Bookwell

Rantala, Ari 2005. Web-ohjelmointi. Porvoo: WS Bookwell

Seybold seminars. Why tables for layout is stupid, problems defined, solutions offered

[verkkodokumentti] [viitattu: 25.4.2010]

Saatavissa: <http://www.hotdesign.com/seibold/>

Tags used in HTML. [verkkodokumentti] W3C

13.11.1992 [viitattu 23.4.2010]

Saatavissa:

<http://www.w3.org/History/19921103-hypertext/hypertext/WWW/MarkUp/Tags.html>

Visual formatting model [verkkodokumentti] W3C

[viitattu 4.5.2010] Saatavissa: <http://www.w3.org/TR/CSS2/visuren.html>

Web Developer [verkkodokumentti]

[viitattu 28.4.2010]

Saatavissa: <http://chrispederick.com/work/web-developer/>

LIITE 1(1). Leipätekstin sijainti uudessa HTML-rakenteessa

```
<div class="conf clearfix">
  <div id="confSectionsBox" class="clearfix">
    <div class="confBodyBox clearfix">
      <!--Main body-->
      <div class="mainContent">
        <div class="col2">
          <table style="width: 70%;">
            <tbody>
              <tr>
                <td align="justify">
                  <p>
                    <font face="Verdana, Arial, Helvetica,
                      sans-serif" size="-2">
                    </font>
                  </p>
                </td>
              </tr>
            </tbody>
          </table>
        </div>
      </div>
    </div>
  </div>
</div>
```

Selitys: Ylläoleva koodi on Indicon versiosta 0.97b2. Se havainnoi, kuinka monen sisäkkäisen rakenteen sisällä leipäteksti sijaitsee. Vaikka sisin rakenne on samanlainen taulu kuin vanhassakin versiossa, on rakenne silti huomattavasti yksinkertaisempi.

LIITE 1(2). Leipätekstin sijainti vanhassa HTML-rakenteessa

```
<table class="conf" align="center" border="0" cellpadding="0"
cellspacing="1" width="100%">
  <tbody>
    <tr>
      <td class="confBodyBox">
        <table cellpadding="0" cellspacing="0" width="100%">
          <tbody>
            <!--Main body-->
            <tr>
              <td>
                <table align="center" border="0" cellspacing="5">
                  <tbody>
                    <tr>
                      <td colspan="2">
                        <table style="width: 70%;">
                          <tbody>
                            <tr>
                              <td align="justify">
                                <p>
                                  <font face="Verdana, Arial,
                                  Helvetica, sans-serif"
                                  size="-2">
                                    </font>
                                </p>
                              </td>
                            </tr>
                          </tbody>
                        </table>
                      </td>
                    </tr>
                  </tbody>
                </table>
              </td>
            </tr>
          </tbody>
        </table>
      </td>
    </tr>
  </tbody>
</table>
```

Selitys: Ylläoleva koodi on Indicon versiosta 96.2. Se havainnoi, kuinka paljon monimutkaisempi ja laajempi vanhan version HTML-koodi on verrattuna uuteen.

LIITE 2. Havainnekuvat kahden erillisen sivuston ongelmasta

CHEP 2009
21-27 March 2009
Prague

[Home](#)

[Conference webpage](#)
[Overview](#)
[Scientific Programme](#)
[Call for Abstracts](#)
• View my abstracts
• Submit a new abstract
[Timetable](#)
[Contribution List](#)
[Author index](#)
[Book of abstracts \[PDF\]](#)
[Streaming](#)

[support](#)

International Conference on Computing in High Energy and Nuclear Physics

Dates: from 21 March 2009 08:00 to 27 March 2009 13:30
Location: Prague
Prague Congress Centre
5. května 65, 140 00 Prague 4, Czech Republic

Chairs: Grun
Loka

Additional info: The ne

CHEP 2009
LOGIN SEARCH

Home
Invitation
General information
Organisation
Programme
Posters
Conference venue
Registration
Abstract submission
Registered participants
Accommodation

Computing in High Energy and Nuclear Physics
Prague | Czech Republic | 21 - 27 March 2009

[Home](#)

CHEP'09
17th International Conference on Computing in High Energy and Nuclear Physics
21 - 27 March 2009 Prague, Czech Republic

Downloads
[Bulletin 1 \(PDF\) \(0.3 MB\)](#)
[Bulletin 2 \(PDF\) \(0.4 MB\)](#)
[Bulletin 3 \(PDF\) \(0.3 MB\)](#)

Strings 2008
17-23 August 2008
CERN

[Home](#)

[Strings 2008](#)
[Registration](#)
• Registration Form
[Participants](#)

[support](#)

See [here](#) for the main web site.

Dates: from 17 August 2008 09:30 to 23 August 2008 13:45

STRINGS2008 Aug 18-23

[News](#) [Talks](#) [Time table](#)

WELCOME

GENERAL INFO
Welcome
News
Info sheets
Organization
Sponsors
Participants
Posters
Past String Conf.
Contact

PROGRAM
Talks
Time table
Social Events
Public Event